



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

NÁVRH DATABÁZE PRO DĚTSKOU SKUPINU

DESIGN OF DATABASE FOR CHILDREN'S GROUP

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Ondřej Vizner

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Luhan, Ph.D., MSc

BRNO 2019

Zadání bakalářské práce

Ústav: Ústav informatiky
Student: Ondřej Vizner
Studijní program: Systémové inženýrství a informatika
Studijní obor: Manažerská informatika
Vedoucí práce: Ing. Jan Luhan, Ph.D., MSc
Akademický rok: 2018/19

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává bakalářskou práci s názvem:

Návrh databáze pro dětskou skupinu

Charakteristika problematiky úkolu:

Úvod
Cíle práce, metody a postupy zpracování
Teoretická východiska práce
Analýza současného stavu
Vlastní návrhy řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Navrhnout databázový systém pro evidenci informací o dětské skupině dle požadavků zadavatele. Zpracovat kompletní návrh z hlediska konceptuálního, logického a fyzického návrhu databáze pro zefektivnění administrativních úkonů zadavatele.

Základní literární prameny:

AUER, D. J. a D. M. KROENKE. Databáze. 1. vyd. Brno: Computer Press, 2015. 496 s. ISBN 978-80-251-4352-0.

BASL, J. a R. BLAŽÍČEK. Podnikové informační systémy - podnik v informační společnosti. 3. vyd. Praha: Grada Publishing, 2012. 328 s. ISBN 978-80-247-4307-3.

BEGG, C., R. HOLOWCZAK a T. CONOLLY. Mistrovství - Databáze : Profesionální průvodce tvorbou efektivních databází. 1. vyd. Brno: Computer Press, 2009. 584 s. ISBN 978-80-251-2328-7.

STEPHENS, R. K., R. PLEW a A. D. JONES. Naučte se SQL za 28 dní: stačí hodina denně. 1 vyd.
Brno: Computer Press, 2010. 728 s. ISBN 978-80-251-2700-1.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2018/19

V Brně dne 28.2.2019

L. S.

doc. RNDr. Bedřich Půža, CSc.
ředitel

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

Abstrakt

Tato bakalářská práce pojednává o návržení databázového systému pro brněnskou dětskou skupinu. Cílem práce je umožnit prostřednictvím navrženého systému zvýšení efektivity práce s daty zejména s ohledem na interní procesy dětské skupiny.

Klíčová slova

ER – diagram, SŘBD, databáze, databázový systém, SQL.

Abstract

This bachelor thesis deals with a topic of designing a database system for a children's group from Brno. The main aim of this thesis is to increase the efficiency of data usage in regard to internal processes within the childrens's group.

Key words

ER – diagram, DBMS, database, database system, SQL.

Bibliografická citace

VIZNER, Ondřej. *Návrh databáze pro dětskou skupinu* [online]. Brno, 2019 [cit. 2019-05-10]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/119573>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Jan Luhan.

Čestné prohlášení

Prohlašuji, že předložená bakalářská práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 10. května 2019

.....

Ondřej Vizner

Poděkování

Rád bych touto cestou moc poděkoval zejména vedoucímu mé bakalářské práce Ing. Janu Luhanovi, Ph.D., MSc za obětavou pomoc a cenné odborné rady při tvorbě mé práce. Dále bych chtěl poděkovat celé mé rodině za velikou podporu, která mi hodně pomohla.

OBSAH

ÚVOD	12
CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ	13
1 TEORETICKÁ VÝCHODISKA PRÁCE.....	14
1.1 Základní pojmy	14
1.1.1 Informace	14
1.1.2 Data.....	15
1.1.3 Znalosti	15
1.2 Databázový systém	16
1.3 Databáze.....	16
1.4 Systém řízení databáze	17
1.4.1 Složky prostředí SŘBD	18
1.4.2 Architektura SŘBD	18
1.5 Databázová aplikace	19
1.5.1 Pohledy	20
1.6 Datové modely.....	20
1.6.1 Lineární datový model.....	21
1.6.2 Objektový datový model	21
1.6.3 Hierarchický datový model.....	22
1.6.4 Síťový model	23
1.6.5 Relační datový model.....	23
1.7 Relace	24
1.7.1 Integrita relačního modelu	25
1.7.2 Integritní omezení pro relace	25
1.7.3 Integritní omezení pro vztahy	26
1.8 Normalizace	28

1.8.1	První normální forma	28
1.8.2	Druhá normální forma	28
1.8.3	Třetí normální forma	28
1.9	SQL.....	29
1.9.1	Kategorie příkazů jazyka SQL.....	29
1.9.2	Jazyk DDL.....	29
1.9.3	Jazyk DQL.....	29
1.9.4	Jazyk DML.....	30
1.9.5	Jazyk DCL.....	30
1.9.6	Snapshot.....	30
1.10	GDPR	31
1.10.1	GDPR zásady zpracování osobních údajů	32
1.10.2	GDPR práva subjektu údajů	33
1.10.3	GDPR zabezpečení osobních údajů.....	35
2	ANALÝZA SOUČASNÉHO STAVU	36
2.1	Obecné informace o dětské skupině	36
2.2	Formální údaje dětské skupiny	36
2.3	Hardwarové vybavení	37
2.4	Softwarové vybavení.....	37
2.5	Evidence klientů	37
2.5.1	Export evidence klientů do Excelu.....	42
2.6	Evidence zaměstnanců	43
2.7	Evidence docházky klientů.....	43
2.8	Nedostatky dětské skupiny	44
3	VLASTNÍ NÁVRHY ŘEŠENÍ.....	45
3.1	Koncepční návrh.....	45

3.1.1	Identifikace entit	45
3.1.2	Identifikace relací.....	47
3.2	Logický návrh.....	49
3.2.1	Tabulka zaměstnanec	49
3.2.2	Tabulka psc.....	50
3.2.3	Tabulka pozice.....	50
3.2.4	Tabulka kontakt zaměstnance	51
3.2.5	Tabulka oprávnění	51
3.2.6	Tabulka pobočka.....	52
3.2.7	Tabulka klient	52
3.2.8	Tabulka Otec dítěte	53
3.2.9	Tabulka kontakt otce dítěte.....	54
3.2.10	Tabulka matka dítěte	54
3.2.11	Tabulka kontakt matky dítěte.....	55
3.2.12	Tabulka platba	55
3.2.13	Tabulka délka docházky	55
3.2.14	Tabulka Přihláška.....	56
3.2.15	Tabulka vyzvedávání.....	58
3.2.16	Tabulka alergie	58
3.2.17	Tabulka rodný list dítěte	58
3.2.18	Tabulka očkovací průkaz.....	59
3.2.19	Tabulka týdenní docházky	59
3.2.20	Tabulka obědy	59
3.2.21	Dekompoziční tabulka počet docházky	60
3.2.22	E-R diagram	60
3.1	Fyzický návrh.....	62

3.1.1	Vytvoření databáze	62
3.1.2	Vytvoření tabulek.....	62
3.1.3	Spojení tabulek pomocí klíčů.....	63
3.1.4	Naplnění tabulky daty	64
3.2	Pohledy	65
3.2.1	Pohled klientů na jednotlivých pobočkách	65
3.2.2	Pohled Zaměstnanců na pobočkách.....	67
3.2.3	Pohled oprávnění zaměstnanců	67
3.2.4	Docházka klientů	68
3.2.5	Pohled přihláška.....	69
3.3	Snapshot.....	70
3.4	Trigger smazání zaměstnance	70
3.5	Zhodnocení návrhu	70
ZÁVĚR		72
SEZNAM POUŽITÝCH ZDROJŮ		73
SEZNAM OBRÁZKŮ		75
SEZNAM TABULEK.....		77
SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ		78
SEZNAM PŘÍLOH.....		79

ÚVOD

Žijeme v době, kdy každodenní rutinou je práce s počítačem. Počítače používáme jak v práci, tak při trávení volného času. Počítače se staly nedílnou součástí našich životů. Stejně tak nedílnou součástí většiny firem.

Technologie jdou neustále dopředu a téměř každá firma si může ulehčit administraci a správu tvorbou databázového systému. Firma díky databázi může lépe evidovat informace o svých zákaznících, o zboží na skladě, o množství zboží, které za kalendářní rok prodala či o dodavatelích.

Je na každé firmě, jestli se vydá směrem, který tu vládnul před padesáti lety, a to evidování všeho na papír nebo jestli půjde vstříc světu technologii a rozhodne se vybudovat vlastní databázi. Je na každé firmě, jakým způsobem se rozhodne zpracovat databázi, které informace chce evidovat a které, evidovat nebude. Každá firma se může rozhodnout, zda zvolí některý z komerčních databázových systémů, či využije nabídku bezplatných možností jako, MySQL.

CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

Hlavním cílem je tvorba databázového systému pro dětskou skupinu působící v Bystřici, který pomůže majitelům zlepšit administraci a správu ve firmě. Primárním cílem je zjistit jaké informace jsou pro majitele důležité a jak by je chtěly evidovat v databázovém systému. Sekundárním cílem je vytvoření návrhu pomocí E-R diagramu, tak aby byla dětská skupina schopná pracovat se vstupními daty.

Dílními cíli práce jsou:

- zpracování teoretické stránky dané problematiky pomocí knižních a jiných zdrojů,
- zpracování analýzy současného stavu ve firmě, zjistit s jakými daty dětská skupina pracuje a s jakými daty bude chtít v databázovém systému pracovat,
- navržení vlastního řešení této problematiky,
- tvorba pohledů a triggerů, které usnadní správu dětské skupiny,
- navrhnout možnou zálohu dat, se kterými bude dětská skupina pracovat.

V této práci byly vytvořeny jednotlivé postupy, které vedli k zdárnému dokončení této práce. Prvním postupem bylo osobní setkání s vedením dětské skupiny. Pomocí této schůzky jsme si vysvětlili, jaký bude výstup mé práce, s jakými daty dětská skupina pracuje, jaké osobní údaje chtějí evidovat. Dalším postupem bylo navrhnutí databázi složenou s konceptuálního, logického a fyzického návrhu.

1 TEORETICKÁ VÝCHODISKA PRÁCE

Tato kapitola se zabývá teoretickými východisky důležitými pro pochopení problematiky této práce. Jako první si v kapitole připomeneme základní pojmy, důležité při práci s databází. Následně se v kapitole budu věnovat tématu databáze a rozeberu z čeho je databáze složená. Dále vysvětlím pojem pohledy a shrnu jednotlivé datové modely a rozeberu téma relace. Následně popíšu jednotlivé normální formy a rozeberu co to je SQL. Na závěr se budu věnovat snapshotům a tématu GDPR.

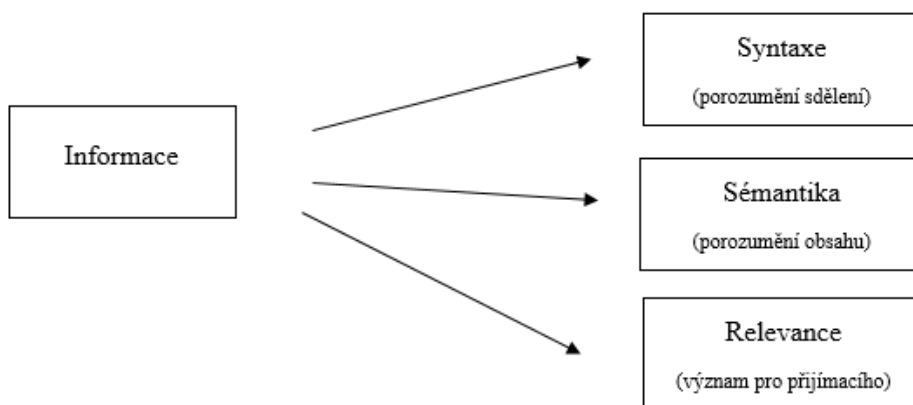
1.1 Základní pojmy

Pro porozumění datového modelování je důležité porozumět pojmům: data, informace, znalosti (1 s. 11).

Jak uvádí Drucker: „*Znalosti a informace jsou dnes jediným smysluplným zdrojem. Tradiční výrobní faktory – půda, práce, kapitál nezmizely, ale staly se druhořadými. Hlavním producentem bohatství jsou informace a znalosti*“ (1 s. 11).

1.1.1 Informace

Informaci lze chápat jako zprávu, která splňuje tři požadavky. Prvním je syntaktická relevance. Druhým požadavkem je sémantická relevance a třetím požadavkem je pragmatická relevance. Všechny tři požadavky zobrazuje **obrázek č. 1** (1 s. 11).

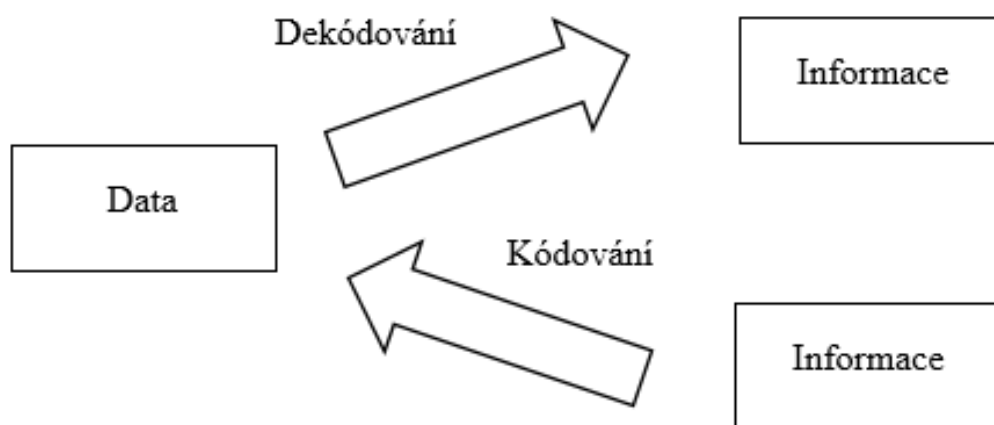


Obrázek č. 1: Znázornění informací (Zdroj: Vlastní zpracování dle: 1, s.4)

- Syntaktická relevance znamená, že přijímací subjekt musí zprávu detekovat a rozumět ji.
- Sémantická relevance znamená, že přijímací subjekt musí vědět co zpráva znamená a co vypovídá o jeho okolí a o něm.
- Pragmatická relevance znamená, že pro přijímací subjekt musí mít zpráva nějaký význam (1, s. 4).

1.1.2 Data

Datům je připisován význam zpráv. Pokud člověk data zrovna používá, stávají se z dat informace, protože se datům přiřazuje význam a smysl. Někdy se stane, že datům je přiřazován nejen význam zpráv, ale také informace. Lze tedy říci, že data jsou potenciální informace. Znázornění lze vidět na **obrázku č. 2** (1, s. 12).



Obrázek č. 2: Znázornění dat (Zdroj: Vlastní zpracování dle: 1, s. 5)

1.1.3 Znalosti

Robert M. Hayes: „*Znalosti jsou výsledkem porozumění informace, která byla právě sdělena, a její integrace s dřívějšími informacemi*“ (1, s. 12).

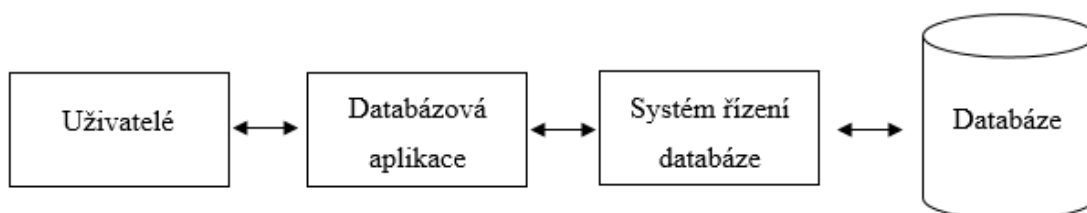
Znalosti lze charakterizovat, jako informace o tom, jak využít informace, data a jejich vzájemné kombinace (1, s. 12).

1.2 Databázový systém

Databázový systém zahrnuje čtyři komponenty:

- Uživatelé,
- databázovou aplikaci,
- systém řízení databáze (DBMS),
- vlastní databázi (3, s. 29).

Databáze obsahuje tabulky a jiné struktury, které spolu navzájem souvisí. Systém řízení databáze je počítačový program, který zpracovává, vytváří a spravuje databázi. Systém řízení databáze je řízen pomocí jazyka SQL, v tomto jazyce přijímá požadavky, které převádí na aktivity v databázi. Databázová aplikace se skládá z programů, které propojují uživatele se systémem řízení dat. Aplikační programy pracují s databázovými daty tak, že odesílají SQL příkazy systému řízení databáze. Aplikační programy pomocí formulářů a sestav prezentují data. Uživatelé si udržují přehled o určitých skutečnostech v databázi pomocí databázové aplikace. Uživatelé zadávají a vytvářejí data, vytvářejí sestavy a dotazují se na ně pomocí formulářů. Na **obrázku č. 3** jsou zobrazeny komponenty databázového systému (3, s. 29).



Obrázek č. 3: Komponenty databázového systému (Zdroj: Vlastní zpracování dle: 3, s. 30)

1.3 Databáze

Databázi můžeme chápat jako velké uložení dat, které mohou používat různí oddělení uživatelé současně. Data, které uživatelé potřebují, jsou integrována, tak aby došlo k minimálním duplikacím. Databázi obvykle nevlastní uživatelé, ale je sdíleným zdrojem společnosti. Databáze obsahuje data dané organizace a dále také popis těchto dat, tzv. metadata. Metadata - „data o datech“ – se označují jako slovník dat nebo systémový katalog (4, s. 37).

Popisující charakter databáze poskytuje tzv. nezávislost dat. To znamená, že pokud změníme nebo přidáme data do existující databáze, tak aplikace, které užívají databázi zůstávají nedotčeny. Pokud závisí na tom, co bylo změněno, tak se v dané aplikaci provede změna. Například pokud odstraníme sloupec z tabulky, který používá daná aplikace (4, s. 37).

Tabulka č. 1: Obsah databáze (Zdroj: Vlastní zpracování dle: 3, s. 30)

Obsah databáze
Uživatelská data
Meta data
Indexy a další režijní informace
Aplikační metadata

1.4 Systém řízení databáze

Systém řízení databáze má za úkol spravovat, vytvářet a zpracovávat databáze. Systém řízení databáze se většinou pořizuje formou licence od dodavatele softwaru, jelikož se jedná o velký a složitý produkt. Jednotlivé systémy řízení dat lze vidět v **tabulce č. 2** (3, s. 31).

Tabulka č. 2: Systémy řízení databáze (Zdroj: Vlastní zpracování dle: 3, s. 31)

Systémy řízení databáze
Microsoft Access
Microsoft SQL Server
MySQL od Oracle Corporation
Oracle Database od Oracle Corporation
DB2 společnosti IBM

Systém řízení databáze umí vytvářet pomocné struktury společně s tabulkami a celými databázemi. Dalšími funkcemi systému řízení databáze je úprava a čtení dat a také funkce související se správou databáze (3, s. 31).

Další funkce SŘBD:

- Vytvoření databáze,
- vytvoření tabulek,
- vytvoření podpůrných struktur,
- čtení dat z databáze,
- úpravy databázových dat,
- údržba databázových struktur,
- vynucování pravidel,
- kontrola souběžnosti,
- zajištění bezpečnosti,
- záloha a obnova dat (3, s. 31).

1.4.1 Složky prostředí SŘBD

Můžeme definovat pět hlavních složek prostředí SŘBD, jedná se o hardware, software a dále pak data, která fungují jako můstek mezi hardwarovou a softwarovou složkou a lidskou složkou. Databáze obsahují jak provozní data, tak metadata (4, s. 41).

Další složkou jsou procedury, jedná se o instrukce a pravidla, která řídí návrh a používání databáze. Jedná se například o instrukce, jak vytvořit záložní kopii a jak zvládnout selhání softwaru a hardwaru databáze (4, s. 41).

Poslední složkou jsou osoby, které spravují databázi nebo aplikační programátoři a uživatelé (4, s. 41).

1.4.2 Architektura SŘBD

Architektury SŘBD se dělí na dvě části před vznikem webu:

1. Program klienta, který se stará o obchod a procesní logiku spojenou s obchodem, dalším úkol programu klienta je vytvoření rozhraní pro uživatele.
2. Program serveru, který se nazývá DBMS engine, který zaručuje správu a řízení postupu k databázi (4, s.41).

Další architektura SŘBD je známá pod názvem dvouvrstvá architektura, která je spojením architektury klienta a serveru. Tato architektura měla problém se složitějšími aplikacemi, které byly rozmístěny až k tisícům koncovým uživatelům. Díky tomu klientská strana způsobovala vzniklé problémy spojené s:

- „Tlustým“ klientem, který vyžadoval velké zdroje v klientském počítači, aby mohl efektivně fungovat.
- Zatížení klienta administrací (4, s.42).

Kvůli těmto problémům vznikla třívrstvá architektura. Tato architektura má tři vrstvy a každá z vrstev může fungovat na jiné platformě. Jednotlivé vrstvy jsou:

1. Vrstva uživatelského rozhraní, která funguje u koncového uživatele (4, s. 43).
2. Vrstva zpracování dat a logiku provozu. Tato střední vrstva funguje na serveru. Aplikační server byl navržen tak, aby mohl obsluhovat více klientů (4, s. 43).
3. Vrstva SŘBD , která je navržena, tak aby uchovávala data, které požaduje střední vrstva. Tato vrstva může běžet na oddělených serverech (4, s. 43).

Třívrstvá architektura má oproti dvouvrstvým modelům výhody, těmito výhodami jsou například:

- „Tenký“ klient, díky kterému nemusíme mít nákladný hardware.
- Zvýšená modularita, která podporuje nahrazení jedné vrstvy bez ovlivnění ostatních vrstev.
- Snadnější údržba aplikací, díky centralizaci provozní logiky. To zamezuje distribuce softwaru, která je problematická u dvouvrstvé architektury.
- Vyrovnaná zátěž, díky oddělení jádra provozní logiky od databázových funkcí.
- Třívrstvá architektura odpovídá prostředí webu, s prohlížečem jako „tenkým“ klientem a webovým serverem (4, s. 43).

1.5 Databázová aplikace

Každý uživatel pracuje s databází pomocí databázových aplikací, které vytváří a spravují databázi a generují informace. Programy mohou být obvyklými aplikacemi nebo online aplikacemi. Databázové aplikace můžeme psát pomocí programovacích jazyků třetí generace, jako například C++ nebo Java anebo v programovacím jazyku čtvrté generace. Další funkce databázové aplikace obsahuje **tabulka č. 3** (3, s. 32).

Tabulka č. 3: Funkce databázových aplikací (Zdroj: Vlastní zpracování dle 3, s. 32)

Funkce databázových aplikací
Vytvoření a zpracování formulářů
Zpracování uživatelských dotazů
Vytvoření a zpracování sestav
Provádění aplikační logiky
Řízení aplikace

1.5.1 Pohledy

Jedná se o virtuální tabulku, která v databázi většinou neexistuje, ale generuje ji systém řízení báze dat z tabulek, které se nachází v aktuální databázi, na kterých je pohled založen a kdykoliv se k ní přistupuje (4, s. 41).

Pohled je dotaz, který pracuje a zpracovává tabulky, aby mohl vytvořit virtuální tabulku. Kromě výhody, že uživatel vidí data právě tak, jak si přeje, tak mají pohledy další výhody:

- Pohledy poskytují zabezpečení.
- Pohledy poskytují mechanismus.
- Pohled může vytvořit neměnný obraz struktury databáze (4, s. 41).

1.6 Datové modely

Pokud chceme vytvořit informační systémy, tak nevystačíme s jedinou strukturou věty.

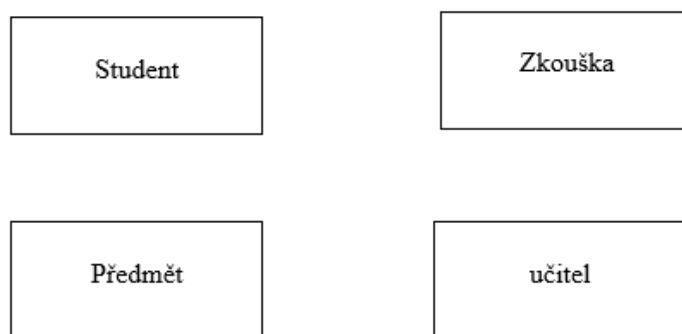
Při tvorbě databáze lze pro použití v zásadě zvolit jeden z pěti možných typů datových modelů. Jde o datové modely:

- Lineární,
- hierarchický,
- síťový,
- relační,
- objektový (1 s. 21).

1.6.1 Lineární datový model

Na **obrázku č. 4** je zachycen lineární datový model. V lineárních datových modelech nenajdeme vazbu mezi jednotlivými skupinami objektů – tabulkami. Jedná se o datový model, který můžeme implementovat na libovolné médium (2, s. 21).

Příkladem lineárního modelu je kartotéka pacientů u lékaře, kde jsou seřazeny v zásuvkách karty s údaji o pacientovi. Karta představuje jednu větu databázového souboru. Mezi ostatními kartami neexistuje žádný vzájemný vztah, pokud nepočítáme vztah předchůdce a následovníka (2, s. 21).

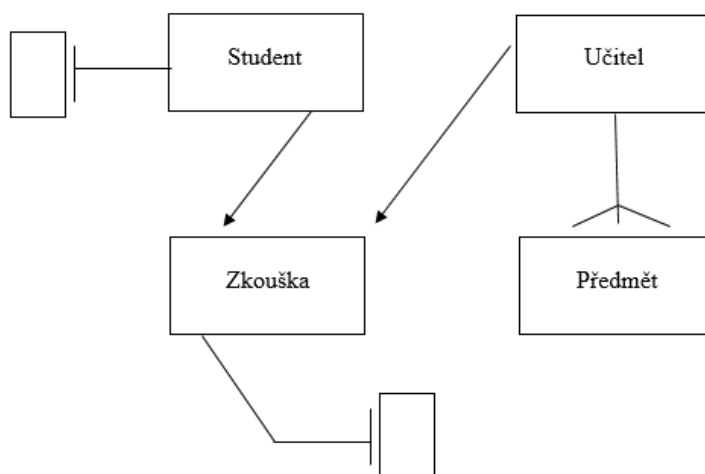


Obrázek č. 4: Lineární datový model (Zdroj: Vlastní zpracování dle: 2, s.21)

1.6.2 Objektový datový model

Patří k nejnovějším datovým modelům, objektový datový model se skládá z prvků (objektů), kde tento objekt obsahuje kromě svých modelů i definované metody, které určují chování objektů. Pokud je objektem „zkouška“ studenta, tak atributem objektu jsou údaje o známce, datu, předmětu, termínu a zkoušejícím. Objekt „zkouška“ může mít definované metody, například metodu „vytvoř záznam o zkoušce“ kde se zkontroluje, zda se nepřekročila kapacita dané zkoušky nebo jestli nebyl vyčerpán studentův poslední termín (2, s 22).

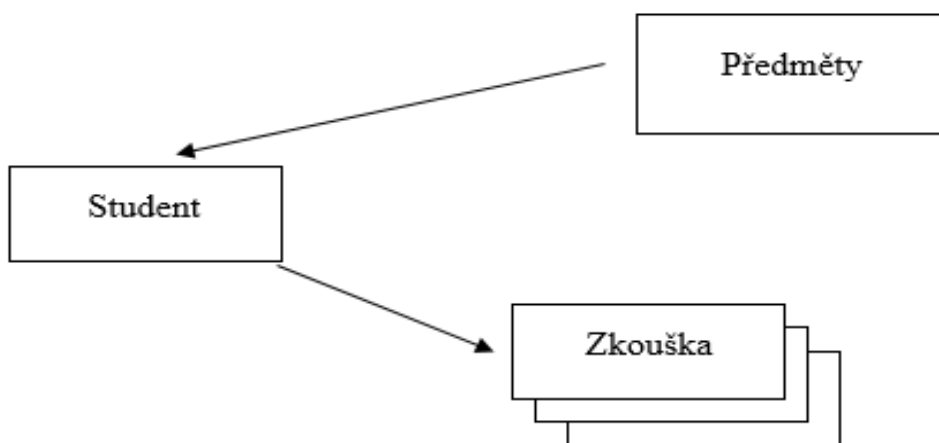
Více objektů, které mají stejný typ tvoří třídu objektů, například třída studentů. Konkrétní záznamy o studentech nazýváme instance objektu. V databázi má každý objekt přidělen unikátní identifikátor, díky kterému můžeme mezi objekty vést vazby jako v síťovém modelu. Příklad objektového datového modelu můžeme vidět na **obrázku č. 5** (2, s. 22).



Obrázek č. 5: Objektový datový model (Zdroj: Vlastní zpracování dle: 2, s.22)

1.6.3 Hierarchický datový model

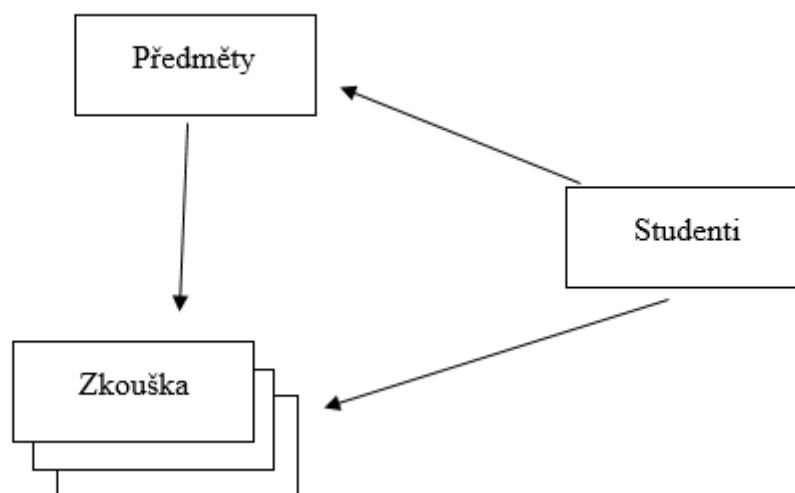
Hierarchický datový model je tvořen rodičovskou větou, ze které vedou vazby na podřízené větě, což jsou větě, které jsou tvořené jinou strukturou a obsahem. Jestliže budeme mít rodičovskou větu s údaji o studentovi, pak jeho podřízené segmenty budou obsahovat informace o uskutečněných zkouškách. Záznamů bude tolik, kolika zkoušek se student zúčastnil. Mezi jednotlivými segmenty existují takzvané „pointery“, které vytváří databázový systém. Na podřízené segmenty se nelze dostat jinak než přes rodičovský segment. Na **obrázku č. 6** je znázorněno schéma modelu (1, s. 22).



Obrázek č. 6: Hierarchický datový model (Zdroj: Vlastní zpracování dle: 2, s.22)

1.6.4 Síťový model

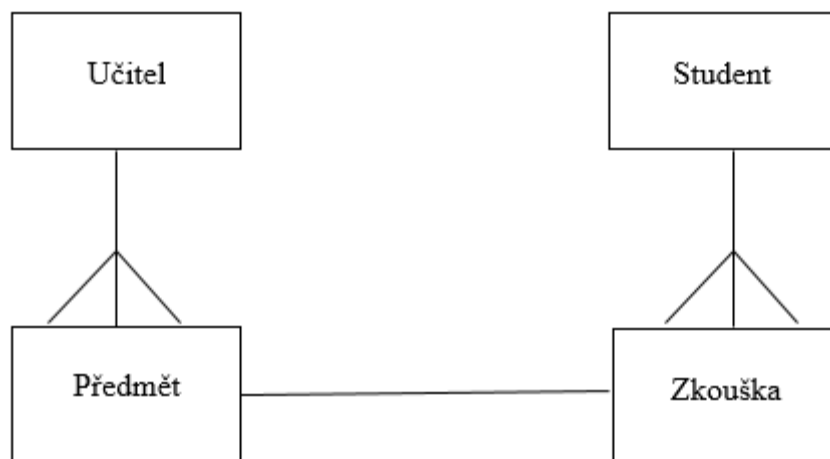
Síťový model je podobný jako hierarchický s tím rozdílem, že pointery vedou ne pouze z hlavního segmentu na vedlejší segmenty, ale obecně mezi segmenty databáze v různých směrech. V síťovém modelu nehovoříme o hlavní větě a jejích podřízených větách, ale pouze o segmentech. Šipky v modelu znázorňují vazby mezi větami. Mezi výhody patří libovolné propojení segmentů a rychlý přístup k datům. **Obrázek č. 7** znázorňuje síťový model (1, s. 22).



Obrázek č. 7: Síťový model (Zdroj: Vlastní zpracování dle: 1, s.23)

1.6.5 Relační datový model

Relační datový model patří k nejpoužívanějším datovým modelům. Vzniká spojením několika lineárních modelů spojených dohromady pomocí položek nazývaných relační klíče. Toto spojení není trvalé, ale vzniká v okamžiku, kdy potřebujeme mít k dispozici data ze všech tabulek spojených relací a zaniká v okamžiku, kdy práci s modelem ukončíme. Jednotlivé lineární modely lze využívat i samostatně. Na **obrázku č. 8** je zobrazen relační datový model (2, s. 21).



Obrázek č. 8: Relační datový model (Zdroj: Vlastní zpracování dle: 2, s.21)

Relační datová struktura

Relační model má pět hlavních složek, těmi jsou jmenovitě: **relace** (jedná se o tabulku se sloupci a řádky), **atributy** (pojmenovaný sloupec relace), **datová n-tice** (řádek relace), **doména** (množina přípustných hodnot pro atribut) a **relační databáze** (normalizované tabulky) (3, s. 63).

Informace o objektech, které mají být v databázi reprezentovány, jsou ukládány pomocí relací, které jsou v relačním modelu. Relace je reprezentována pomocí tabulky, ve které řádky odpovídají jednotlivým n-ticím a sloupce odpovídají atributům. Atributy se mohou v databázi objevit v libovolném pořadí. Domény mají důležitou vlastnost v relačním modelu a tou je, že každý atribut je spojen s doménou. Každá doména může být odlišná i v případě více atributů, nebo mohou být dvě domény spojeny do jednoho atributu (3, s. 64).

1.7 Relace

V databázi jsou sledovány takzvané entity, to jsou data, která jsou pro uživatele důležitá, a která chce reprezentovat v databázi. Databáze se ukládají ve formě relací, které se tváří jako tabulky. Relace je tabulka dvou rozměrů, která se skládá ze sloupců a řádků, a která má následující vlastnosti uvedené v **tabulce č. 4** (3, s. 78).

Tabulka č. 4: Vlastnosti relací (Zdroj: Vlastní zpracování dle 3, s. 78)

Vlastnosti relací
Každý sloupec musí mít jedinečný název
Každý sloupec tabulky obsahuje data, která se týkají nějaké entity nebo části této entity.
Buňky v tabulce musí uchovávat jedinou hodnotu.
Všechny položky v každém sloupci musí být stejného druhu.
Všechny sloupce musí mít jedinečný název.
Na pořadí sloupců v tabulce nezáleží.
Nezáleží ani na pořadí řádků.
Sada datových hodnot musí být jedinečná na každém řádku.

1.7.1 Integrita relačního modelu

Pokud chceme modelovat data z reálného světa, tak nám to přináší určitá omezení v teoretickém modelu. Integritu lze chápat jako určitý stav, ve kterém dat uložené odpovídají vlastnostem v reálném světě. Tyto integritní omezení rozlišujeme na:

- Integritní omezení pro relace,
- integritní omezení pro relační vazby (2, s. 27).

1.7.2 Integritní omezení pro relace

Integritní omezení pro relace dělíme na:

- **Doménová integrita (integrita hodnot)**

Každý atribut z relace musí být z množiny hodnot pro daný atribut přípustných:

1. Definice domény jako množiny hodnot.
2. Specifikace povolených hodnot pro daný atribut.
 - Typ pole,
 - rozsah hodnot,

- maska pro vkládání,
- seznam přípustných hodnot,
- implicitní hodnoty,
- jedinečnost hodnot v rámci sloupce (2, s. 28).
- **Entitní integrita**

Pro každou relaci musí být určen tzv. primární klíč, jedná se o jeden nebo více atributů, jejichž hodnoty jednoznačně identifikují každý řádek z relace (2, s. 28).

Primární klíč – množina atributů s těmito vlastnostmi:

- Jednoznačnost – v relaci neexistuje druhá n-tice, která by měla stejné hodnoty pro tuto množinu.
- Minimálnost – žádný atribut není možné vynechat, aniž bychom porušily pravidla jednoznačnosti (2, s. 28).

Kandidátní klíč – má stejné vlastnosti jako primární klíč s tím rozdílem, že není vybrán jako primární klíč. Každá relace může obsahovat více kandidátních klíčů, jeden z nich je vybrán jako primární klíč. Ostatní kandidátní klíče se stávají alternativami. Způsob výběru primárního klíče zůstává na volbě analytika (2, s. 28).

- **Referenční integrita**

Cizí klíč – atribut, který splňuje vlastnosti:

- Každá hodnota je buď plně zadána nebo nezadána.
- Existuje další relace s primárním klíčem, ve které jsou identické hodnoty primárního klíče s hodnotami cizího klíče (2, s. 28).

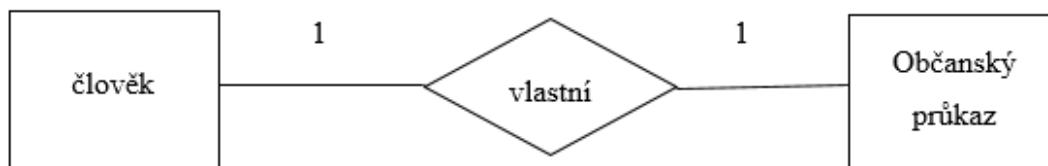
1.7.3 Integritní omezení pro vztahy

Integritní omezení pro vztahy omezuje kardinalitu vztahu na poměry:

1. Vztah 1:1
2. Vztah 1:N
3. Vztah N:1
4. Vztah N:M

Tento poměr uvádí, kolik n-tic sobě odpovídá navzájem (2, s. 30).

- 1) **Vztah 1:1** – Pokud si vztah přiblížíme na příkladu, pak nám říká, že jedna n-tice odpovídá druhé nebo žádné n-tici jiné relace. Příkladem je vztah mezi člověkem a občanským průkazem. Jeden člověk může vlastnit jeden nebo žádný občanský průkaz. Toto zobrazení je na **obrázku č. 9** (2, s. 30).



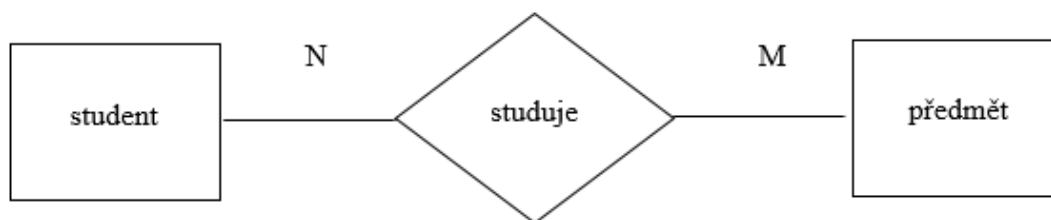
Obrázek č. 9: Vztah 1:1 (Zdroj: Vlastní zpracování dle 2, s. 30)

- 2) **Vztah 1:N** - Pokud si vztah přiblížíme na příkladu, pak nám říká, že jedné n-tici odpovídá více n-tic jiné relace. Příkladem může být vztah mezi člověkem a auty. Jeden člověk může vlastnit jedno nebo více aut a jedno auto patří konkrétnímu člověku. Vztah N:1 je stejný s tím rozdílem, že se na něj koukáme z jiné strany. Vztah 1:N je zobrazen na **obrázku č. 10** (2, s. 32).



Obrázek č. 10: Vztah 1:N (Zdroj: Vlastní zpracování dle 2, s. 32)

- 3) **Vztah N:M** - Pokud si vztah přiblížíme na příkladu, pak nám říká, že více n-tic odpovídá více n-ticím jiné relace. Příkladem může být vztah mezi studenty a jejich studovanými předměty. Více studentů studuje stejný předmět a jeden student studuje více předmětů. Vztah N:M je zobrazen na **obrázku č. 11** (2, s. 32).



Obrázek č. 11: Vztah N:M (Zdroj: Vlastní zpracování dle 2, s. 32)

1.8 Normalizace

Technika, která se používá pro vytvoření několika tabulek s minimální redundancí, která podporuje požadavky dané organizace nebo uživatele. Normalizace se provádí jako testy na dané tabulce, aby bylo prokázáno, že tabulka dodržela nebo porušila pravidla normální formy. Existuje více normálních forem celkem pět, ale nejpoužívanější jsou: první normální forma, druhá normální forma a třetí normální forma, ve své práci použiji první tři formy, proto je blíže popíšu (4, s. 188).

1.8.1 První normální forma

Tabulka splňuje pravidla první normální formy, jestliže v ní každý sloupec obsahuje jen jednu hodnotu. Sloupec nesmí obsahovat složené nebo více hodnotové atributy. První normální forma je jedinou formou, která je nezbytně nutná pro vytvoření tabulek pro relační databázi (4, s. 191).

1.8.2 Druhá normální forma

Tabulka splňuje pravidla druhé normální formy, jestliže splňuje podmínky první normální formy a hodnoty každého sloupce, který není součástí primárního klíče, jsou determinovány všemi hodnotami sloupců, které tvoří primární klíč. Druhá normální forma se týká tabulek, které mají složené primární klíče, tedy tabulek jejichž primární klíč tvoří dva nebo více sloupců (4, s. 192).

1.8.3 Třetí normální forma

Tabulka splňuje pravidla třetí normální formy, jestliže splňuje podmínky první a druhé normální formy. Zároveň musí její neklíčové atributy být závislé výhradně na primárním klíči, a nikoli na sobě navzájem (4, s. 195).

1.9 SQL

Jedná se o standardní jazyk, který používáme při komunikaci s relačními databázemi. SQL je neprocedurální jazyk, to znamená, že počítači sdělíme, jaké výsledky požadujeme, a přitom nemusíme definovat, jak má výsledky získat (5, s. 33).

SQL nepatří do skupiny jazyků jako jsou Pascal, C, Ada, Basic, které jsou procedurálními jazyky. Posloupnost příkazů se musí spouštět v určitém pořadí v procedurálních jazycích. V procedurálních jazycích existují příkazy, které mohou měnit pořadí provádění příkazů (5, s. 33).

1.9.1 Kategorie příkazů jazyka SQL

SQL příkazy se dělí do několika kategorií v závislosti na své funkci. Každá z kategorií se řídí stejnými pravidly a syntaxemi. Proto jednotlivé kategorie můžeme považovat za kategorie stejného jazyka. Jednotlivé kategorie jsou:

- Jazyk DDL,
- jazyk DQL,
- jazyk DML,
- jazyk DCL,
- příkazy řízené transakcí (5, s. 40).

1.9.2 Jazyk DDL

Do jazyka DDL spadají příkazy, které vytváří databázové objekty a upravují jejich strukturu například tabulky, pohledy nebo indexy. Součástí jazyka DDL jsou příkazy, které obsahují klíčová slova CREATE, DROP, ALTER. Příkazy jazyka DDL nemají vliv na vlastní data ale na kontejnery, které uchovávají data v databázi (18, s. 41).

1.9.3 Jazyk DQL

Do jazyka DQL spadají příkazy, které načítají data z databáze. I když se jedná o důležitou součást jazyka SQL, jsou příkazy založeny na jediném klíčovém slově SELECT (18, s. 41).

1.9.4 Jazyk DML

Do jazyka DML spadají příkazy, které umožňují uživatelům přidávat data do databáze, odebírat data z databáze a měnit stávající data. Součástí jazyka DML jsou příkazy, které obsahují klíčová slova UPDATE, DELETE, INSERT (18, s. 41).

1.9.5 Jazyk DCL

Do jazyka DCL spadají příkazy, které dovolují správcům řídit přístup k datům v databázi a používat systémová opatření SŘBD. Tyto oprávnění mohou spustit nebo vypnout databázi. Součástí jazyka DCL jsou příkazy, které obsahují klíčová slova ALTER, GRANT (18, s. 41).

1.9.6 Snapshot

Snapshot je termín pro schopnost zaznamenat stav k danému okamžiku, určený pro rychlou obnovu dat. Snapshot je kopií (point-in-time-copy) k určitému času. Snapshot po vytvoření lze použít k zálohování, testování nebo analýze dat. Originální data jsou nadále dostupná i po vytvoření kopie, a tudíž můžeme pracovat s originálem bez přerušení provozu jen s minimálním pozastavením datového toku. Pomocí snapshotu můžeme obnovit omylem smazaná data bez pomoci obnovy zálohovacího softwaru. Lze vytvořit více snapshotů v krátkém časovém období, to však sebou nese i nároky na výkon a na kapacitu datového úložiště (12).

Snapshoty lze vytvořit pomocí více metod, nejčastější metodou vytváření snapshotu je metoda (copy-on-write), kdy se snapshot vytvoří v prostoru vyhrazený pro snapshoty. Při vytváření snapshotu jsou nejprve zkopírovány metadata s definicí umístění bloků originálních dat. Při změně originálních dat, jsou nejprve data nahrána do prostoru určený pro snapshoty a posléze jsou přepsána (12).

Snapshot je víceúrovňový algoritmus poprvé popsáný v Berensonu. Snapshoty jsou populárnější, protože poskytují úroveň ochrany, která se vyhýbá možným poruchám v databázi. Poprvé byly implementovány společností Oracle a Microsoft SQL Server (11, s. 1).

Pokud má dojít k obnově databáze pomocí snapshotu musí se dbát na omezení, která sebou vytváření zálohy nese a ty jsou:

- Nelze vytvořit snapshot k systémovým databázím.
- Zdrojová databáze i databáze snapshotu je offline v průběhu obnovy.
- Lze vytvořit více snapshotů k jedné databázi, ale při obnově musí existovat pouze jeden (10).

1.10 GDPR

Dětská skupina pracuje s osobními údaji klientů a zaměstnanců, proto tato kapitola bude zaměřena na nová nařízení evropského parlamentu, která vyšli dne 27. dubna 2016 a dne 25. května 2018 přímo stanovují pravidla pro zpracování osobních údajů v EU. Nařízením se musí řídit dětská skupina, která se nazývá správcem osobních údajů, obecným nařízením se musí řídit i zpracovatel, tedy subjekt, který pro správce osobních údajů údaje zpracovává (5).

Celý název je Nařízení Evropského parlamentu a Rady (EU) 2016/679. Jedná se o nařízení o ochraně osobních údajů fyzických osob v souvislosti se zpracováním a o volném pohybu těchto údajů a zrušení směrnice 95/46/ES (5).

Obecné nařízení představuje právní rámec ochrany osobních údajů v evropském prostoru, které 25. květnem 2018 vstupuje v platnost a stanovuje pravidla pro práci s osobními údaji. Nové nařízení nahrazuje v českém právním prostředí zákon č. 101/2000 Sb., o ochraně osobních údajů z roku 1995 (5).

K revizi osobních údajů bylo přikročeno, protože právní rámec 95/46/ES o ochraně fyzických osob v souvislosti se zpracováním osobních údajů, již přestal odpovídat současné době zejména díky prostředkům, které jsou ke zpracování osobních údajů využívány, a které jsou komplexnější než před dvaceti čtyřmi lety. Dalšími důvody revize je nesjednocení právní úpravy směrnice 95/46/ES v jednotlivých zemích Evropské unie (5).

Hlavní rozdíly mezi obecným nařízením o ochraně osobních údajů a směrnicí z roku 1995 jsou:

- Posílení působnosti.
- Posílení ochrany dětí.
- Posílení informačních povinností správců.
- Posílení práv subjektů údajů.
- Jiná výše pokut (6).

1.10.1 GDPR zásady zpracování osobních údajů

O zásadě zpracování osobních údajů se můžeme dočíst ve druhé kapitole článku pět až jedenáct v nařízení Evropského parlamentu a Rady (EU) 2016/679 dále budu používat zmiňovanou zkratku GDPR (13).

Osobní údaje musí být:

- Zpracovány korektně a zákonným a transparentním způsobem vzhledem k subjektu.
- Shromažďovány pro určité, výslovně vyjádřené a legitimní účely a nesmějí být zpracovány způsobem, který je s uvedenými způsoby neslučitelný. Osobní údaje mohou být shromažďovány pro účely vědeckého či historického výzkumu nebo pro statistické účely, které se dle čl. 89 odstavce 1 nepožadují za způsoby neslučitelné s původními účely.
- Musí být zpracovány v rozsahu nezbytném k účelu, pro který jsou zpracovány, dále musí být omezené, relevantní a přiměřené.
- Osobní údaje, které jsou nepřesné musí být vymazány nebo opraveny.
- Uloženy ve formě, která umožňuje identifikovat subjekt údajů po dobu ne delší, než je nezbytné pro účely, pro které jsou zpracovány.
- Uložit způsobem, který zajistí zabezpečení osobních údajů subjektu, včetně ochrany před neoprávněným či protiprávním zpracováním a před ztrátou či zničením nebo poškozením (13).

Ve druhé kapitole se lze dočíst o povinnostech Správce systému, který odpovídá za uvedené body ve článku šest v prvním odstavci a musí být schopen tyto tvrzení dodržení souladu doložit. Odstavec jedna obsahuje, za jakých podmínek lze zákoně zpracovávat osobní údaje. Pro dětskou skupinu platí níže uvedený bod, který je platný pouze v plném rozsahu:

- „*Subjekt údajů udělil souhlas pro zpracování svých osobních údajů, pro či více konkrétních účelů*“ (13).

Článek sedm ve druhé kapitole GDPR stanovuje podmínky vyjádření souhlasu pro zpracování osobních informací. Správce systému se musí řídit těmito pravidly:

1. Pokud je zpracování založeno na souhlasu, musí správce systému doložit, že subjekt údajů udělil souhlas se zpracováním (13).
2. Pokud je souhlas vyjádřen písemným prohlášením, musí být žádost předložena způsobem, který je srozumitelný, jasný a za použití jednoduchých jazykových prostředků (13).
3. Subjekt údajů má právo svůj souhlas kdykoliv odvolat. Před udělením souhlasu musí být subjekt informován o možnosti odvolání souhlasu (13).
4. Souhlas musí být svobodný (13).

Článek devět ve druhé kapitole stanovuje podmínky pro uložení osobních údajů pro zvláštní kategorie, které jsou:

„Zakazuje se zpracování osobních údajů, které vypovídají o rasovém či etnickém původu, politických názorech, náboženském vyznání či filozofickém přesvědčení nebo členství v odborech, a zpracování genetických údajů, biometrických údajů za účelem jedinečné identifikace fyzické osoby a údajů o zdravotním stavu či o sexuálním životě nebo sexuální orientaci fyzické osoby“ (13).

Jelikož dětská skupina pracuje se zvláštní kategorií osobních údajů a těmi jsou alergie a zdravotní stav dítěte, které se v dětské skupině evidují pro případ životně důležitých zájmů subjektu.

Víše zmíněný citovaný odstavec se nepoužije v dětské skupině za předpokladu, že:

- Subjekt údajů udělil výslovný souhlas se zpracováním těchto údajů.
- Zpracování je nutné pro ochranu životně důležitých zájmů subjektu, nebo jiné osoby, která není fyzicky nebo právně způsobilá udělit souhlas.
- Zpracování je nezbytně nutné z důvodů veřejného zájmu v oblasti veřejného zdraví (13).

1.10.2 GDPR práva subjektu údajů

Každý, kdo poskytne nějaký osobní údaj, tak má svá práva, která si rozebereme v této kapitole. Jelikož dětská skupina pracuje též s osobními údaji, tak by měla vědět práva svých klientů k poskytování osobních informací která eviduje.

O jednotlivých právech se lze dočíst ve třetí kapitole oddílu jedna až pět článku dvanáct až dvacet dva v nařízení Evropského parlamentu a Rady (EU) 2016/679.

Oddíl jedna více zmiňovaného nařízení zmiňuje transparentnost a postupy, které musí správce systému poskytnout. Oddíl dva zmiňuje informace a přístupy k osobním údajům, které si blíže popíšeme (13).

Článkem třináct se budeme řídit, pokud jsou osobní údaje získány od subjektu údajů, v našem případě od zaměstnanců a klientů. Tento článek stanovuje správci systému v okamžiku získání osobních údajů tyto povinnosti, které musí subjektu sdělit:

- Totožnost a kontakt na správce systému a jeho zástupce.
- Kontaktní údaje případného pověřence pro ochranu osobních údajů.
- Účely zpracování a právní základ pro zpracování.
- Příjemce nebo kategorie příjemců osobních údajů.
- Doba, po kterou budou osobní údaje uloženy.
- Existence práva požadovat od správce právo na úpravu nebo výmaz, jejich osobních údajů
- Existence práva na podání stížnosti u dozorového úřadu.
- Skutečnost, zda je poskytování osobních údajů zákonným nebo smluvním požadavkem (13).

Článek patnáct v druhém oddílu stanovuje práva subjektu údajů na přístup k osobním údajům. Daný subjekt, kterému se zpracovávají osobní údaje má právo o získání informací, zda jsou jeho údaje zpracovány a dále má právo získání přístupu k těmto osobním údajům a k následujícím informacím, které mají povinnosti správci sdělit subjektu v článku čtrnáct (13).

Oddíl tři stanovuje podmínky pro opravu a výmaz. V případě dětské skupiny mohou mít zaměstnanci a klienti právo na to, aby správce bez odkladu opravil nebo vymazal dané osobní údaje, které se ho týkají, pokud je splněn jeden z těchto důvodů:

- Osobní údaje se už nepoužívají pro účely, které byli při získávání údajů.
- Subjekt údajů odvolá souhlas, jestliže neexistuje další důvod pro zpracování.
- Jestliže neexistují převažující oprávněné důvody pro zpracování a uložení osobních údajů.
- Osobní údaje byly zpracovány protiprávně.
- Osobní údaje musí být vmazány pro splnění právní povinnosti stanovené Unii nebo členským státem, které se na správce vztahuje (13).

Dále má správce povinnosti vymazat osobní údaj bez ohledu na náklady a technologie s tím spojené, dále je povinen zajistit, aby informaci o výmaz dostaly všichni správci, kterým byly poskytnuty tyto osobní údaje, musí být dále odstraněny všechny kopie a replikace daného údajů (13).

Dále má subjekt údajů na právo omezení zpracování údajů ze strany správce. Tyto důvody jsou popsány v článku osmnáct třetího oddílu třetí kapitoly. Dále má správce povinnost oznámit subjektům, že jejich osobní údaje byli opraveny nebo vymazány (13).

1.10.3 GDPR zabezpečení osobních údajů

Zabezpečení musí provést správce a zpracovatel, kteří budou mít vhodná technická a organizační opatření, aby byla zajištěna úroveň zabezpečení odpovídající danému riziku, případě včetně:

- Pseudonymizace a šifrování osobních údajů.
- Zajistit neustálou důvěrnost, integritu, dostupnost a odolnost služeb zpracování.
- Zajistit obnovu dostupnosti osobních údajů a přístup k nim v případě incidentu.
- Zajistit proces testování, hodnocení, posuzování zavedených opatření pro zajištění bezpečnosti zpracování (13).

Jakékoliv porušení zabezpečení musí správci osobních údajů neodkladně do 72 hodin od okamžiku, kdy se o porušení zabezpečení dozví, oznámit dozorovanému úřadu, ledaže je nepravděpodobné, že porušení nebude mít riziko pro práva a svobody fyzických osob. Pokud se incident neohlásí do 72 hodin, musí se uvést důvod proč došlo ke zpoždění. Pokud je pravděpodobnost rizika pro práva a svobody fyzických osob, musí správce danou situaci nahlásit vlastníkům osobních údajů bezodkladně od dané situace (13).

2 ANALÝZA SOUČASNÉHO STAVU

Následující kapitola je věnována představení dětské skupiny, do které bude implementován informační systém. Nejdříve budou popsány obecné informace o dětské skupině. V dalších krocích pak také hardware a software, který dětská skupina používá. V závěru kapitoly budou také shrnuty nedostatky vyplývající z analýzy, které pomohou při sestavení návrhové části.

2.1 Obecné informace o dětské skupině

Dětská skupina Filiorum působí na trhu od roku 2016. Za tu dobu se přesunula její provozovna z ulice Nad Kašnou, do ulice Klácelova, kde převzala dětskou skupinu Kulíšek. Nyní se chystá další změna provozovny do ulice Tvrdého.

Dětská skupina se zabývá výhradně hlídáním dětí od jednoho roku. Dále dětem poskytuje cvičení na speciálních míčích a pravidelně pro děti hraje loutkové divadlo. Dětská skupina dále pronajímá prostory, pro různé kroužky.

Dětská skupina je zapojená do projektu celé Česko čte dětem, které rozvíjí u dětí fantazii a paměť. Pro rozvoj slovní zásoby používá dětská skupina Šimonovy pracovní listy.

2.2 Formální údaje dětské skupiny

Formální název:

Rodinné centrum FILIORUM, z.s.

Obchodní značka dětské skupiny:

Rodinné centrum FILIORUM, z.s.

Právní forma:

Spolek

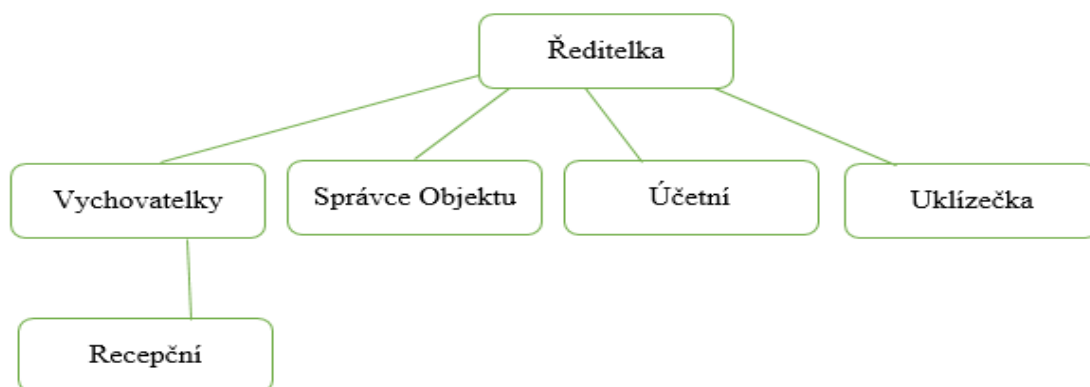
Sídlo:

Nad kašnou 348/30, Bystrc, 635 00 Brno

Datum zápisu:

21. 12. 2016

Organizační struktura firmy:



Obrázek č. 12: Organizační struktura firmy (Zdroj: Vlastní zpracování, dle 14)

2.3 Hardwarové vybavení

Dětská skupina používá jeden notebook, jeden stolní počítač a jednu tiskárnu. Na stolním počítači, který se nachází v provozovně, je uložena docházka vychovatelek a dětí a dále je zde počítač zapojený k tiskárně. Dětská skupina využívá tiskárnu, pro tisk různých materiálů potřebných ke každodenní činnosti. Notebook používá majitelka ke správě dětské skupiny a ke komunikaci s klienty. Na notebooku je nainstalován účetní program POHODA.

2.4 Softwarové vybavení

Dětská skupina používá převážně kancelářský balíček MS Office, především MS Excel, ve kterém je zpracovávána docházka dětí a vychovatelek. Dále také používá MS Word k tvorbě pracovních listů.

Účetnictví je zpracováno pomocí účetního programu POHODA.

Emailová komunikace funguje přes Mozilla Thunderbird a gmail.

2.5 Evidence klientů

Evidence klientů probíhá převážně prostřednictvím tří papírových formulářů, které dostanou všichni zájemci o služby dětské skupiny.

V prvním formuláři se vyplňuje jméno a příjmení dítěte, rok narození a trvalé bydliště. Dále pak také základní informace o matce a otci dítěte, jako např. telefonní číslo, email, místo trvalého pobytu. Příklad tohoto formuláře lze vidět na **obrázku č. 13**.



**ŽÁDOST O PŘIJETÍ DÍTĚTE
DO DS FILIORUM
POBOČKA KLÁCELOVA 2 BRNO**

Rodinné centrum FILIORUM, z.s.

Žádám o přijetí dítěte do DS FILIORUM – pobočka Klácelova 2, Brno 602 00

Od 12.12.2018

Dítě:

Jméno a příjmení Lucie

Datum narození 11.7.

Místo trvalého pobytu Matzernaurova

Zákonní zástupci dítěte:

Otec dítěte

Jméno a příjmení Marlin

Datum narození 13.11.

Místo trvalého pobytu Šaščína

Telefon. číslo 739 648

e-mail —

Obrázek č. 13: Evidence klienta (Zdroj: Vlastní zpracování, dle 15)

Druhým vyplňovaným formulářem, který klienti při přihlášení do dětské skupiny musí odevzdat je tzv. zmocnění osob. V tomto formuláři jsou vypsány osoby, které jsou zmocněny vyzvedávat dítě z dětské skupiny. Osoby, kterým zákonní zástupci neudělí zmocnění na vyzvedávání dítěte, nemohou dítě z dětské skupiny odvádět. Příklad tohoto formuláře lze vidět a **obrázku č. 14.**



ZMOCNĚNÍ OSOB

Rodinné centrum FILIORUM, z.s.

Jméno dítěte Lucie

Zákonní zástupci dítěte:

Otec dítěte Martin

Matka dítěte Michaela

Udělují tímto plnou moc níže uvedeným osobám k vyzvedávání dítěte z DS
FILIORUM – pobočka Klácelova 2, Brno

Jméno a příjmení Vladimíra

Datum narození 24.1.

Místo trvalého pobytu Hatzenauerova

Jméno a příjmení Michal

Datum narození 24.9.

Místo trvalého pobytu Hatzenauerova

Jméno a příjmení

Datum narození

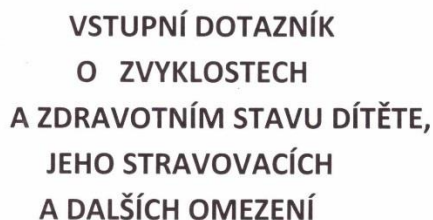
Místo trvalého pobytu

V Brně 12.12.2018

podpis zákonného zástupce

Obrázek č. 14: Zmocnění osob (Zdroj: Vlastní zpracování, dle 16)

Ve třetím formuláři se vyplňují alergie dítěte, jeho koníčky a záliby. Tyto informace slouží zejména pro větší přehled pracovníků dětské skupiny. Příklad tohoto formuláře lze vidět na **obrázku č. 15**.



Jméno a příjmení dítěte Lukáš
Datum narození 12.7. Zdravotní pojišťovna 111
Oslovení dítěte zdobnělinou Lukáško, Lukáš (NE Lukčo!)
Osoby, na které je dítě fixováno na nikoho je to zaměřeno

Oblíbené činnosti (zábava) čtení knih, malování, zpívání, cvičení, oblékání, hra s hračkami, hračky a pohádkami

Je dítě schopno samo si říct, že chce jít na WC? (zakroužkujte) ANO / NE

Je dítě schopno samo si říct, že něco chce, že něco potřebuje? ANO / NE

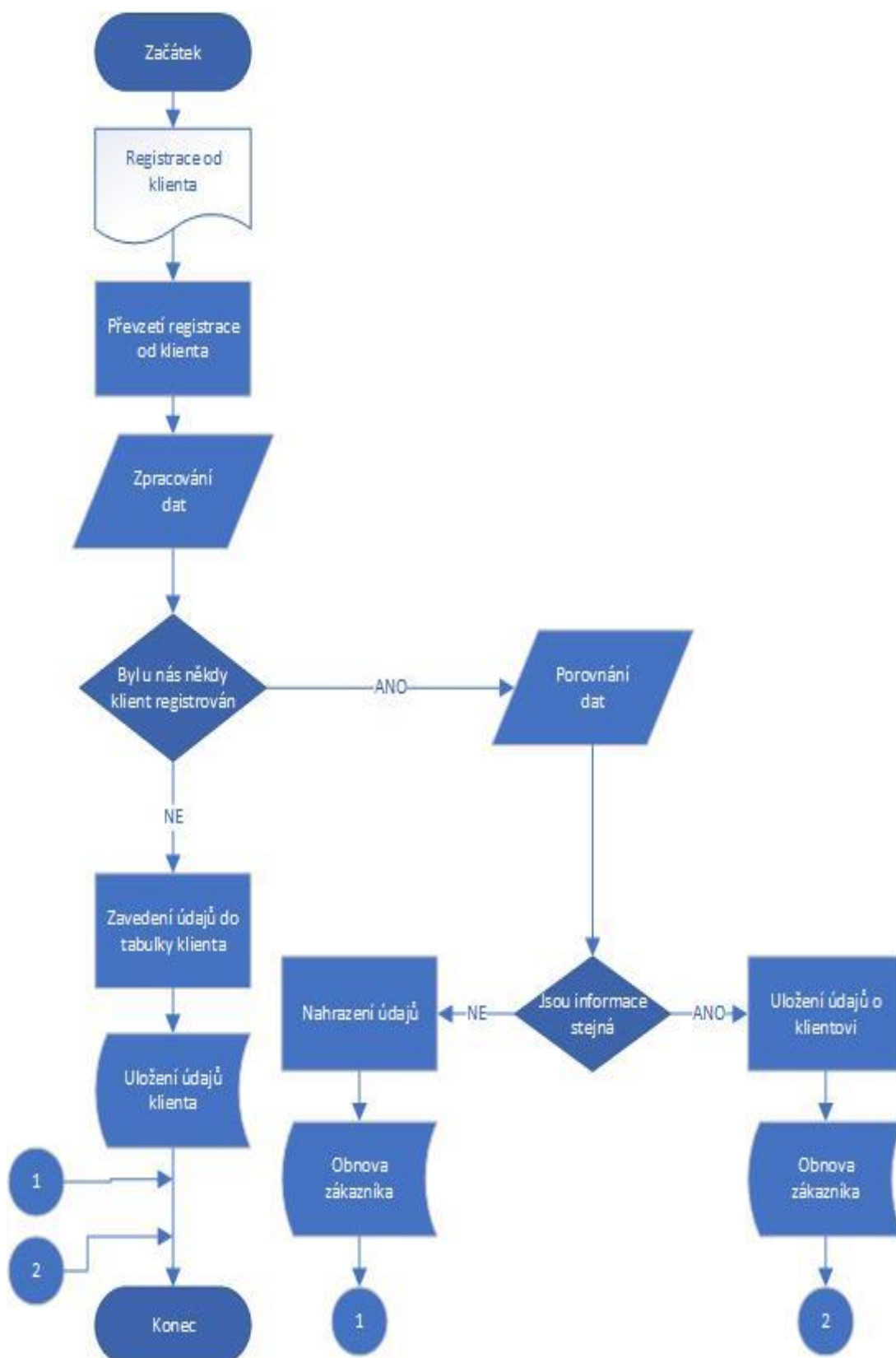
Domlouvá se specifickým způsobem? (popište) ne

Je zvyklé pravidelně odpočívat? V jakou dobu? (popište)
ne, děti, ale už se jí moc nechce (maminka chová)

Je něco, čeho se bojí, něco co špatně přijímá? (popište)
ne, čertí (2 děti strach)

- Dítě je zdravé a může být přijato do Dětské skupiny FILIORUM bez omezení
- Dítě vyžaduje speciální péči v oblasti/oblastech

K výše zmíněným formulářům je také přikládán očkovací průkaz a rodný list dítěte. Po odevzdání formulářů se všechny formuláře naskenují, uloží v počítači a na externím hardisku. Následně se tyto informace přepíší i do Excelu, který umožňuje jejich rychlejší vyhledávání.



Obrázek č. 16: Evidence klientů diagram (Zdroj: Vlastní zpracování)

2.5.1 Export evidence klientů do Excelu

Aby informace o klientech nebyly pouze na papíře, jsou informace z formulářů přepisovány také do MS Excel. Dětská skupina používá celkem tři excelové tabulky. První eviduje klienty, kteří aktuálně chodí do dětské skupiny, druhá tabulka eviduje budoucí klienty, kteří by chtěli využít služby dětské skupiny a poslední tabulka eviduje klienty, kteří již nevyužívají služby dětské skupiny.

První tabulka o klientech eviduje informace, které rodiče vyplnili v prvním formuláři. Jedná se o údaje o jménu a příjmení dítěte a adrese dítěte. Dále obsahuje informace o matce a otci dítěte, zda rodiče dodali rodný list dítěte a očkovací průkaz. V neposlední řadě tabulka obsahuje jméno a telefon osoby, která může dítě vyzvedávat, a také poznámku do které se zapisuje, zda dítě např. netrpí poruchou chování nebo nemá alergie. Ukázku této tabulky znázorňuje **obrázek č. 17**.

Jméno a příjmení dítěte	Datum narození	Adresa	Matka	Adresa	Email	Telefon	Otec	Adresa	Email	Telefon	Očkování	Rodný list	Zmocnění osob	Telefon	Poznámka
Petr Novák	31.7.2017	Krajní 15	Jana Nováková	Krajní 15	novakova@seznam.cz	736458963	Jakub Novák	Krajní 15	Novák@seznam.cz	736412896	ANO	ANO	Oba rodiče	736458963	
Krystýna Polní	20.2.2016	Drdy 20	Jana Polní	Drdy 20	janapolni@gmail.com	789456123	neuveden	neuvedena	neuveden	neuveden	ANO	ANO	Matka, babička	789456123	
Zuzana Kábrtová	10.2.2016	Výhon 3	Anežka Kábrtová	Výhon 3	kabrtova@gmail.com	741852963	Ondřej Kábrt	Výhon 3	neuveden	796385241	ANO	ANO	rodiče, babička	741852369	Alergie citrusy
Hana Stolařová	9.1.2015	Dědina 7	Tereza Dvořáková	Dědina 20	dvorakova@gmail.com	741852364	Jan Stolař	Pustiny 77	stolar@gmail.com	763258941	ANO	ANO	Matka	741852364	
Anežka Lipovská	25.3.2015	Krajní 30	Pavla Lipovská	Krajní 30	lipovska@gmail.com	741148529	Tomáš Lipovský	neuvedena	neuveden	neuveden	ANO	ANO	rodiče, babička	741148529	
Tomáš Stálý	28.7.2016	Česká 30	Zdislava Stálá	Česká 30	stala@seznam.cz	741258852	Filip Stálý	Česká 30	staly@seznam.cz	789456123	ANO	ANO	rodiče, babička	789456123	Alergie ořechy
Jan Novák	23.7.2016	Krajní 15	Jana Nováková	Krajní 15	novakova@seznam.cz	736458963	Jakub Novák	Krajní 15	Novák@seznam.cz	736412896	ANO	ANO	Oba rodiče	736458963	
Anna Lipská	20.2.2016	Drdy 20	Zuzana Lipská	Drdy 30	lipska@gmail.com	789456123	neuveden	neuvedena	neuveden	neuveden	ANO	ANO	Matka, babička	789456123	
Alena Kábrtová	17.4.2014	Výhon 3	Anežka Kábrtová	Výhon 3	kabrtova@gmail.com	741852963	Ondřej Kábrt	Výhon 3	neuveden	796385241	ANO	ANO	rodiče, babička	741852369	Alergie citrusy
Zbírka Potočná	9.2.2015	Dědina 30	Tereza Potočná	Dědina 30	dvorakova@gmail.com	741852364	Jan Hybl	Pustiny 100	stolar@gmail.com	763258941	ANO	ANO	Matka	741852364	
Anežka Krajní	25.7.2015	Česká 77	Pavla Krajní	Česká 77	krajni@gmail.com	741148529	Tomáš Krajní	neuvedena	neuveden	neuveden	ANO	ANO	rodiče, babička	741148529	
Jan Stálý	28.7.2015	Česká 30	Zdislava Stálá	Česká 30	stala@seznam.cz	741258852	Filip Stálý	Česká 30	staly@seznam.cz	789456123	ANO	ANO	rodiče, babička	789456123	Alergie ořechy

Obrázek č. 17: Tabulka o současných klientech (Zdroj: Vlastní zpracování, fiktivní údaje)

Druhá tabulka, kterou dětská skupina používá je o klientech, kteří by chtěli v budoucnu využívat služby dětské skupiny. V této tabulce je evidováno: jméno a příjmení dítěte, datum narození, jméno a příjmení rodiče, telefon a email, a také předpokládané datum nástupu do dětské skupiny. Ukázku této tabulky znázorňuje **obrázek č. 18**.

Jméno a příjmení dítěte	Datum narození	Adresa	Matka	Adresa	Email	Telefon	Otec	Adresa	Email	Telefon	Datum nástupu
Petr Novák	31.7.2017	Krajní 15	Jana Nováková	Krajní 15	novakova@seznam.cz	736458963	Jakub Novák	Krajní 15	Novák@seznam.cz	736412896	3.7.2019
Jana Polní	20.2.2016	Drdy 20	Jana Polní	Drdy 20	janapolni@gmail.com	789456123	neuveden	neuvedena	neuveden	neuveden	4.7.2019
Monika Kábrtová	10.2.2016	Výhon 3	Anežka Kábrtová	Výhon 3	kabrtova@gmail.com	741852963	Ondřej Kábrt	Výhon 3	neuveden	796385241	15.8.2019
Hana Stolařová	9.1.2015	Dědina 7	Tereza Dvořáková	Dědina 20	dvorakova@gmail.com	741852364	Jan Stolař	Pustiny 77	stolar@gmail.com	763258941	20.8.2019
Zdislava Lipovská	25.3.2015	Krajní 30	Pavla Lipovská	Krajní 30	lipovska@gmail.com	741148529	Tomáš Lipovský	neuvedena	neuveden	neuveden	25.8.2019
Tomáš Stálý	28.7.2016	Česká 30	Zdislava Stálá	Česká 30	stala@seznam.cz	741258852	Filip Stálý	Česká 30	staly@seznam.cz	789456123	30.8.2019
Štěpán Novák	23.7.2016	Krajní 15	Jana Nováková	Krajní 15	novakova@seznam.cz	736458963	Jakub Novák	Krajní 15	Novák@seznam.cz	736412896	4.9.2019
Anna Lipská	20.2.2016	Drdy 20	Zuzana Lipská	Drdy 30	lipska@gmail.com	789456123	neuveden	neuvedena	neuveden	neuveden	9.9.2019
Alena Kábrtová	17.4.2014	Výhon 3	Anežka Kábrtová	Výhon 3	kabrtova@gmail.com	741852963	Ondřej Kábrt	Výhon 3	neuveden	796385241	14.9.2019
Zbírka Potočná	9.2.2015	Dědina 30	Tereza Potočná	Dědina 30	dvorakova@gmail.com	741852364	Jan Hybl	Pustiny 100	stolar@gmail.com	763258941	19.9.2019
Anežka Krajní	25.7.2015	Česká 77	Pavla Krajní	Česká 77	krajni@gmail.com	741148529	Tomáš Krajní	neuvedena	neuveden	neuveden	24.9.2019
Jan Stálý	28.7.2015	Česká 30	Zdislava Stálá	Česká 30	stala@seznam.cz	741258852	Filip Stálý	Česká 30	staly@seznam.cz	789456123	29.9.2019

Obrázek č. 18: Tabulka o budoucích klientech (Zdroj: Vlastní zpracování, fiktivní údaje)

Třetí tabulka eviduje klienty, kteří již nevyužívají služby dětské skupiny. O těchto klientech se eviduje: jméno a příjmení dítěte, rok narození, jméno a příjmení rodičů dítěte, email, telefon na rodiče a datum, kdy dítě přestalo chodit do dětské skupiny, společně s důvodem odchodu. Ukázku této tabulky znázorňuje **obrázek č. 19**.

Jméno a příjmení dítěte	Datum narození	Adresa	Matka	Adresa	Email	Telefon	Otec	Adresa	Email	Telefon	Datum ukončení	Důvod
Jana Novotná	17.2.2015	Novotná 15	Romana Novotná	Novotná 15	novotna@seznam.cz	789456123	Roman Novotný	Novotná 15	novotny@gmail.com	789456123	18.2.2018	stáří
Dana Holušová	20.6.2014	Kavčí 3	Iveta Holušová	Kavčí 3	holusova@gmail.com	789456123	Ivan Holuš	Kavčí 3	holusova@gmail.com	789456123	19.2.2018	stáří
Anna Lipská	20.2.2016	Drdy 20	Zuzana Lipská	Drdy 30	lipska@gmail.com	789456123	neveden	nevedena	neveden	neveden	20.2.2018	stáří
Alena Kábrtová	17.4.2014	Výhon 3	Anežka Kábrtová	Výhon 3	kabrtova@gmail.com	741852963	Ondřej Kábrt	Výhon 3	neveden	796385241	21.2.2018	stáří
Zbiřka Potočná	9.2.2015	Dědina 30	Tereza Potočná	Dědina 30	dvorakova@gmail.co	741852364	Jan Hybl	Pustiny 100	stolaf@gmail.com	763258941	22.2.2018	stáří
Petr Novák	31.7.2017	Krajní 15	Jana Nováková	Krajní 15	novakova@seznam.cz	736458963	Jakub Novák	Krajní 15	Novák@eznam.cz	736412896	23.2.2018	stáří
Tomáš Stálý	28.7.2016	Česká 30	Zdislava Stálá	Česká 30	stala@seznam.cz	741258852	Filip Stálý	Česká 30	staly@seznam.cz	789456123	24.2.2018	stáří
Jan Novák	23.7.2016	Krajní 15	Jana Nováková	Krajní 15	novakova@seznam.c	736458963	Jakub Novák	Krajní 15	Novák@eznam.cz	736412896	25.2.2018	stáří
Jan Stálý	28.7.2015	Česká 30	Zdislava Stálá	Česká 30	stala@seznam.cz	741258852	Filip Stálý	Česká 30	staly@seznam.cz	789456123	26.2.2018	stáří
Krystýna Polní	20.2.2016	Drdy 20	Jana Polní	Drdy 20	janapolni@gmail.co	789456123	neveden	nevedena	neveden	neveden	30.3.2017	stáří
Zuzana Kábrtová	10.2.2016	Výhon 3	Anežka Kábrtová	Výhon 3	kabrtova@gmail.com	741852963	Ondřej Kábrt	Výhon 3	neveden	796385241	31.3.2017	stáří
Hana Stolařová	9.1.2015	Dědina 7	Tereza Dvořáková	Dědina 20	dvorakova@gmail.co	741852364	Jan Stolař	Pustiny 77	stolaf@gmail.com	763258941	1.4.2017	stáří
Anežka Lipovská	25.3.2015	Krajní 30	Pavla Lipovská	Krajní 30	lipovska@gmail.com	741148529	Tomáš Lipovský	nevedena	neveden	neveden	2.4.2017	stáří

Obrázek č. 19: Tabulka o minulých klientech (Zdroj: Vlastní zpracování, fiktivní údaje)

2.6 Evidence zaměstnanců

Evidence docházky zaměstnanců probíhá „papírovou formou“. Docházka se eviduje na každý měsíc zvlášť. Majitelka do konkrétního sloupečku s příslušným dnem zapíše odchod a příchod zaměstnance. Tyto údaje jsou pak poskytnuty účetní, aby je mohla zpracovat a poslat zaměstnancům jejich mzdu.

2.7 Evidence docházky klientů

Evidence docházky do dětské skupiny probíhá výhradně „papírovou“ formou. Docházka se eviduje na každý týden zvlášť. Rodiče nahlašují, které dny jejich dítě do dětské skupiny dorazí společně s předpokládaným časem odchodu, aby vedení vědělo s kolika dětmi mohou, na jaký den počítat. Pro případ, že rodič neočekávaně dovede svoje dítě do školky bez nahlášení, obsahuje sloupec pro konkrétní den více políček.

Když dorazí dítě do dětské skupiny, je zapsáno jeho jméno a k němu se vyplní i čas příchodu dítěte. Následně se ještě do kolonky služba zapíše, jestli dítě pojede při procházce v kočárku a jestli bude po obědě spát nebo ne. Dále se doplní, jestli rodič docházku dítěte zaplatil. Rezervace dítěte se píše červenou a nečekaná docházka černou, aby to bylo možné rozlišit. Při odchodu dítěte se opět zapisuje v kolik hodin dítě odešlo. Ukázku docházky znázorňuje **obrázek č. 20**.

Rezervace a přehled docházky dětí

v týdnu od 26.12. 2018 do 31.12. 2018

čas příchodu - odchodu	7:00-8:00	8:00-9:00	9:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	16:00-17:00	17:00-18:00	18:00-19:00	19:00-20:00	20:00-21:00	21:00-22:00	22:00-23:00	23:00-24:00
služba / platba	K	I	S	S	S	S	K	D	I	I	D						
pondělí	TOMAS	PETE	JITKA	PAUL	ANŽELKA	LUCIA	TOMAS	LUCIA									
čas příchodu - odchodu	7:00-8:00	8:00-9:00	9:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	16:00-17:00	17:00-18:00	18:00-19:00	19:00-20:00	20:00-21:00	21:00-22:00	22:00-23:00	23:00-24:00
služba / platba	D	S	S	S	S	D	D	D	D	I	I	S					
úterý	TOMAS	PETE	JITKA	PAUL	ANŽELKA	LUCIA	TOMAS	LUCIA	DAN	IVAN	IVETA						
čas příchodu - odchodu	7:00-8:00	8:00-9:00	9:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	16:00-17:00	17:00-18:00	18:00-19:00	19:00-20:00	20:00-21:00	21:00-22:00	22:00-23:00	23:00-24:00
služba / platba	D	S	S	S	S	D	I	S									
středa	JITKA	PETE	ANŽELKA	LUCIA	PAUL	IVAN	IVETA										
čas příchodu - odchodu	7:00-8:00	8:00-9:00	9:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	16:00-17:00	17:00-18:00	18:00-19:00	19:00-20:00	20:00-21:00	21:00-22:00	22:00-23:00	23:00-24:00
služba / platba	D	S	S	D	S	S	S	I									
čtvrtek	DAN	JITKA	TOMAS	ANŽELKA	PETE	DAN	LUCIA										
čas příchodu - odchodu	7:00-8:00	8:00-9:00	9:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	16:00-17:00	17:00-18:00	18:00-19:00	19:00-20:00	20:00-21:00	21:00-22:00	22:00-23:00	23:00-24:00
služba / platba	D	D	S	S	S	S	S										
pátek	JITKA	ANŽELKA	DAN	IVAN	IVETA	PAUL	LUCIA										

Obrázek č. 20: Evidence docházky (Zdroj: Vlastní zpracování, dle listu z DS, fiktivní údaje)

2.8 Nedostatky dětské skupiny

Mezi největší nedostatky dětské skupiny jistě patří neexistence uceleného systému, pomocí, kterého se usnadní správa dětské skupiny. Pokud dojde k situaci, kdy budou jednotlivé formuláře zničeny, tak nedojde k obnově, protože dětské skupině chybí jakákoliv záloha, která by zabezpečovala informace o klientech a zaměstnancích.

Dalším nedostatkem je problém s evidencí docházky jednotlivých klientů. Docházka se eviduje pouze na papírový formulář, pokud si budou chtít majitelé spočítat docházku klienta s určitým jménem za daný měsíc, tak budou muset projít čtyři papírové formuláře, ve kterých budou hledat dotyčné jméno a příjmení. To sebou nese řadu možných negativních situací, které mohou nastat, jako je přehlédnutí klienta a spočítání mu kratší docházky, než ve skutečnosti dítě odchodilo.

Dalším nedostatkem je chybějící formulář s kontakty na rodiče dětí v dětské skupině. Kontakty na rodiče klientů jsou pouze v přihlášce a ve firemním mobilu ale pokud dojde k poruše firemního mobilu a zaměstnanec bude chtít zavolat rodiči od některého z dětí, tak musí najít jednotlivou přihlášku, až pak zjistit daný kontakt.

3 VLASTNÍ NÁVRHY ŘEŠENÍ

Tato kapitola se zabývá vlastním návrhem řešení, v rámci kterého bude vytvořen konceptuální, logický a fyzický návrh databáze, který pomůže řešit nedostatky, které vyplynuli v předchozí kapitole. Mezi tyto nedostatky patří např. lepší správa dat dětské skupiny.

Postupně budou identifikovány jednotlivé entity potřebné pro správný chod databázového systému společně s jejich relacemi. Následně budou blíže specifikovány veškeré tabulky databázového systému, včetně informací o jednotlivých entitách a dalších náležitostech, jako jsou např. datové typy jednotlivých atributů. Na základě těchto poznatků bude vytvořen ER-diagram systému.

V poslední části této kapitoly bude vytvořen samotný fyzický návrh databáze. Čili jednotlivé tabulky, potřebné pohledy, triggerly a snapshoty. V rámci této kapitoly budou zobrazeny i útržky kódů, které byly využity pro tvorbu jednotlivých částí databáze.

3.1 Konceptuální návrh

V rámci konceptuálního návrhu je nutné vymezit informace, které se budou v systému uchovávat. Je zapotřebí identifikovat jednotlivé entity, které se budou nacházet v systému a také veškeré relace, které budou tyto entity propojovat.

3.1.1 Identifikace entit

Následující tabulka zobrazuje veškeré entity spojené s tvorbou databáze.

Každá entita má svůj název a stručný popis, který vysvětlí, k čemu slouží, respektive jaká data budou v rámci této entity evidována. Dále bude obsahovat předpokládané množství výskytů entity v databázi.

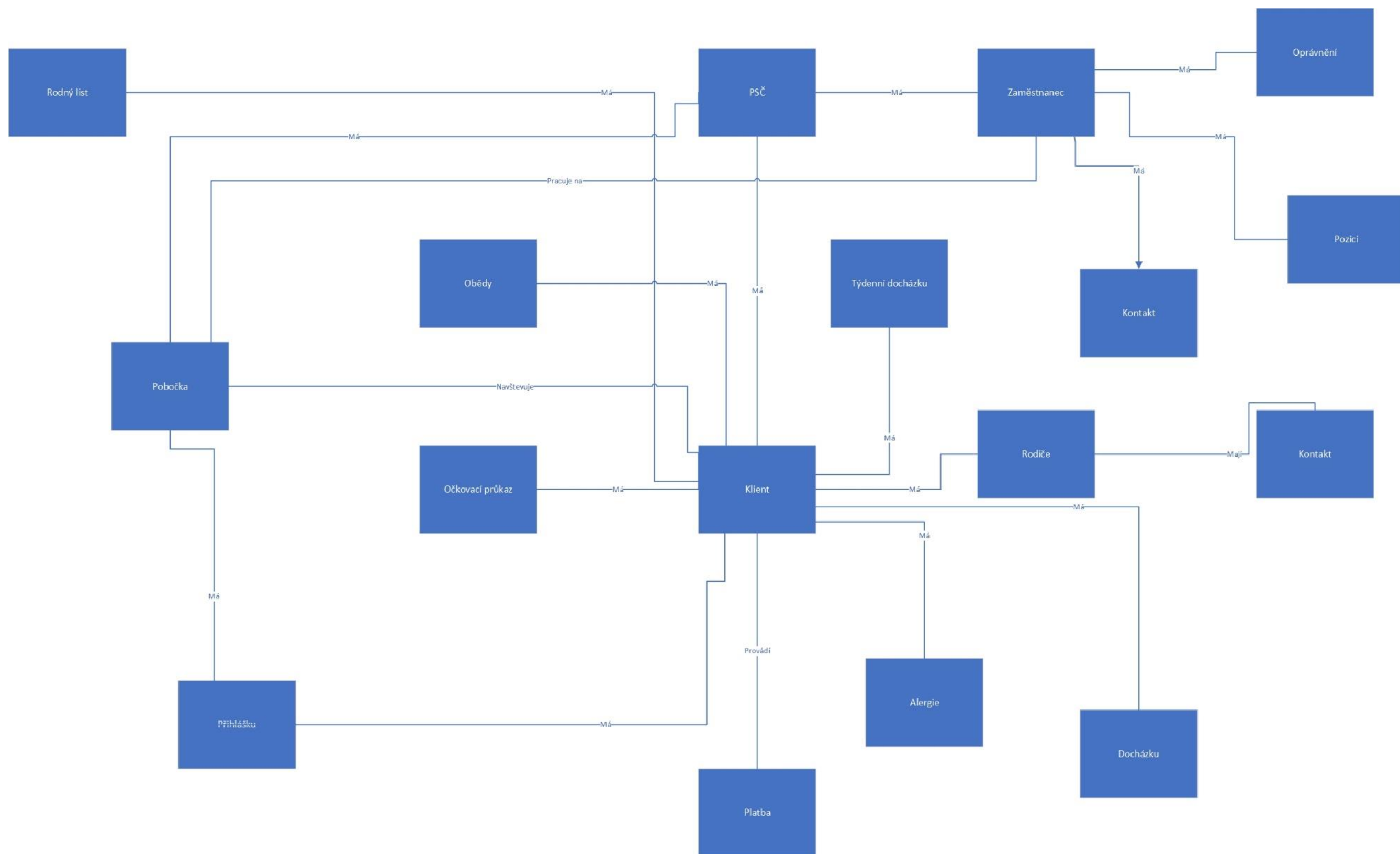
Tabulka č. 5: identifikace entit (Zdroj: Vlastní zpracování)

Název entity	Popis entity	Počet výskytů
Zaměstnanec	Pracovník/ce, který bude pracovat v dětské skupině	Desítky
Kontakt_zam	Eviduje kontaktní údaje zaměstnance dětské skupiny	Desítky
Přihláška	Přihláška do dětské skupiny	Tisíce
Klient	Klient zapsaný do dětské skupiny	Stovky
Matka_ditete	Matka dítěte v dětské skupině	Stovky
Otec_ditete	Otec dítěte v dětské skupině	Stovky
Kontakt_otec	Eviduje kontaktní údaje otce dítěte	Stovky
Kontakt_matka	Eviduje kontaktní údaje otce dítěte	Stovky
PSC	Eviduje poštovní směrovací čísla České republiky	Stovky
Vyzvedávání	Eviduje jména osob, které budou dítě vyzvedávat	Stovky
Platba	Eviduje způsob zaplacení docházky dítěte za daný měsíc, dále obsahuje, zda byl poplatek uhrazen, či nikoli	Tisíce

Týdení_docházka	Eviduje předpokládanou docházku dětí v týdnu.	Tisíce
Délka_docházky	Eviduje dny, ve kterých dítě bude navštěvovat dětskou skupinu, dále délku docházky	Stovky
Pobočka	Eviduje pobočky dětské skupiny	Jednotky
Pocet_dochazky	Dekompoziční tabulka pro práci s docházkou u jednotlivých klientů	Stovky
Pravomoce	Eviduje úroveň oprávnění s prací v systému	Desítky
Pozice	Eviduje pozice zaměstnanců, dětské skupiny	Desítky
Alergie	Eviduje alergie dětí	Stovky
Rodný_list	Eviduje, zda rodič dodal rodný list dítěte	Stovky
Očkovací_průkaz	Eviduje, zda rodič dodal očkovací průkaz dítěte	Stovky
Obědy	Eviduje, zda dítě bude chodit na obědy	Stovky

3.1.2 Identifikace relací

Tato kapitola identifikuje všechny relace mezi tabulkami, které se nachází v návrhu pro dětskou skupinu. Identifikace relací je pro větší přehlednost zpracována pomocí **obrázku č. 21**. Jednotlivé entity jsou zobrazovány pomocí modrých čtverců, mezi kterými se pak nachází jednotlivé relace společně se slovním popisem této relace.



Obrázek č. 21: Identifikace relací (Zdroj: Vlastní zpracování)

3.2 Logický návrh

V rámci logického návrhu si definuji tabulky popsané v kapitole 3.1.1. U každé tabulky si nadefinuji atributy. U všech atributů si určím název, popis, typ a délku, zda musí být atribut vyplněn, a nakonec si určím primární a cizí klíče. Primární a cizí klíče se budou barevně rozlišovat. V tomto případě jsem si vybral pro primární klíč tmavě modrou barvu a pro cizí klíč světle modrou barvu.

3.2.1 Tabulka zaměstnanec

Tato tabulka eviduje informace o jednotlivých zaměstnancích, kteří jsou zaměstnaní pod dětskou skupinou. V **tabulce č. 6** evidujeme osobní informace o jednotlivých zaměstnancích, a také jejich pozice a pravomoc pro informační systém.

Tabulka č. 6: Tabulka zaměstnanec (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_zames	Identifikační číslo zaměstnance (automaticky generované číslo)	Int	Ano
Jmeno	Jméno zaměstnance	Varchar 30	Ano
Prijmeni	Příjmení zaměstnance	Varchar 40	Ano
Titul_pred	Zaměstnancův titul před jménem	Varchar 5	Ne
Titul_zo	Zaměstnancův titul za jménem	Varchar 5	Ne
RC	Rodné číslo zaměstnance	Varchar 11	Ano
Datum_narozeni	Datum narození zaměstnance	Date	Ano
PSC	PSČ bydliště zaměstnance	Int	Ano

Opraveni	Pravomovce zaměstnance	Int	Ano
Kontakt_zam	Kontakt zaměstnance	Int	Ano
Pozice	Pozice zaměstnance	Int	Ano
Pobočka	Pobočka, ve které zaměstnanec pracuje	Int	Ano

3.2.2 Tabulka PSČ

Tato tabulka eviduje konkrétní adresu klienta nebo pobočky nebo zaměstnance, tabulka obsahuje poštovní směrovací čísla s městem klienta a pobočky, společně s jeho ulicí a číslem popisným.

Tabulka č. 7: Tabulka PSC (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_psc	Identifikační číslo psč	Int	Ano
Mesto	Město bydliště	Varchar 30	Ano
PSC	Poštovní směrovací číslo	Varchar 6	Ano
Ulice	Ulice bydliště	Varchar 50	Ano
CP	Číslo popisné adresy	Varchar 2	Ano

3.2.3 Tabulka pozice

Tabulka eviduje jednotlivé pozice zaměstnanců dětské skupiny. V tabulce nalezneme danou pozici a její detailní popis.

Tabulka č. 8: Tabulka pozice (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_pozice	Identifikační číslo pozice zaměstnance	Int	Ano
Pozice	Název dané pozice	Varchar 15	Ano
Popis	Popis pozice zaměstnance	Text	Ano

3.2.4 Tabulka kontakt zaměstnance

Tabulka kontakt zaměstnance eviduje kontaktní telefonní číslo a email na každého zaměstnance dětské skupiny.

Tabulka č. 9: Tabulka zaměstnanec (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_kon_zam	Identifikační číslo kontaktu zaměstnance	Int	Ano
Kontaktní_telefon	Kontaktní telefon zaměstnance	Varchar 16	Ano
email	Kontaktní email zaměstnance	Varchar 50	Ano

3.2.5 Tabulka oprávnění

Tabulka oprávnění eviduje, jakou pravomoc má zaměstnanec s prací s informačním systémem v dětské skupině. Tabulka obsahuje dané pravomoce a jejich popis.

Tabulka č. 10: Tabulka oprávnění (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_opra	Identifikační číslo oprávnění zaměstnance	Int	Ano

Pravomoc	Pravomoc zaměstnance	Varchar 16	Ano
Popis	Popis jednotlivých pravomocí, které mohou mít zaměstnanci.	Text	Ano

3.2.6 Tabulka pobočka

Tabulka pobočka je vytvořena s ohledem do budoucna, kdyby se dětská skupina rozrostla o další pobočku. Tabulka obsahuje adresu dané pobočky společně se zaměstnanci, kteří na pobočce pracují, dále obsahuje jména klientů navštěvující pobočku.

Tabulka č. 11: Tabulka pobočka (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_pobočka	Identifikační číslo pobočky dětské skupiny	Int	Ano
Nazev	Název pobočky	Varchar 30	Ano
PSC	Adresa pobočky	Int	Ano

3.2.7 Tabulka klient

Tabulka klient eviduje veškeré klienty navštěvující dětskou skupinu. Tabulka obsahuje osobní informace o dětech jako je jméno a příjmení, datum narození, adresa a rodné číslo, dále se odkazuje na tabulky svých rodičů a na platbu, aby bylo zřetelné, zda byla zaplacená docházka. Ke každému klientovi se eviduje přihláška.

Tabulka č. 12: Tabulka klient (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_klienta	Identifikační číslo Klienta (automaticky generované číslo)	Int	Ano
Jmeno	Jméno klienta	Varchar 30	Ano

Prijmeni	Příjmení klienta	Varchar 40	Ano
RC	Rodné číslo klienta	Varchar 11	Ano
Datum_narozeni	Datum narození klienta	Date	Ano
PSC	PSC bydliště klienta	Int	Ano
Otec_ditete	Otec dítěte	Int	Ne
Matka_ditete	Matka dítěte	Int	Ne
Platba	Zda klient zaplatil docházku	Int	Ano
Pobočka	Pobočka, kterou klient navštěvuje	Int	Ano

3.2.8 Tabulka Otec dítěte

Tabulka č. 13 eviduje jména a příjmení tatínků od dětí navštěvující dětskou skupinu. Dále se tabulka odkazuje na kontakt, aby bylo tatínka v případě nutnosti možnost kontaktovat.

Tabulka č. 13: Tabulka otec dítěte (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_otec	Identifikační číslo tatínka dítěte	Int	Ano
Jmeno	Jméno otce dítěte	Varchar 35	Ano
Prijmeni	Příjmení otce dítěte	Varchar 40	Ano

Kontakt_otec	Kontakt na otce dítěte	Int	Ano
--------------	------------------------	-----	-----

3.2.9 Tabulka kontakt otce dítěte

Tabulka eviduje kontaktní telefon a email na otce dítěte navštěvující dětskou skupinu.

Tabulka č. 14: Tabulka kontakt otec dítěte (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_kon_otec	Identifikační číslo kontaktu zaměstnance	Int	Ano
Kontaktni_telefon	Kontaktní telefon otce dítěte	Varchar 16	Ano
email	Kontaktní email otce dítěte	Varchar 50	Ano

3.2.10 Tabulka matka dítěte

Tabulka č. 15 eviduje jména a příjmení maminek od dětí navštěvující dětskou skupinu. Dále se tabulka odkazuje na kontakt, aby bylo maminky v případě nutnosti možno kontaktovat.

Tabulka č. 15: Tabulka matka dítěte (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_matka	Identifikační číslo matky dítěte	Int	Ano
Jméno	Jméno matky dítěte	Varchar 35	Ano
Příjmení	Příjmení matky dítěte	Varchar 40	Ano
Kontakt_matka	Kontakt na otce dítěte	Int	Ano

3.2.11 Tabulka kontakt matky dítěte

Tabulka eviduje kontaktní telefon a email na matku dítěte navštěvující dětskou skupinu.

Tabulka č. 16: Tabulka kontakt matky dítěte (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_kon_matka	Identifikační číslo kontaktu matky dítěte	Int	Ano
Kontaktní_telefon	Kontaktní telefon matky dítěte	Varchar 16	Ano
email	Kontaktní email matky dítěte	Varchar 50	Ano

3.2.12 Tabulka platba

Tabulka platba eviduje informace o platbě a zároveň o stavu plateb, který byly provedeny.

Tabulka č. 17: Tabulka platba (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_platba	Identifikační číslo platby	Int	Ano
Zaplateno	Zobrazuje, zda byla docházka zaplacená	Boolean	Ano
Platba	Způsob platby docházky	Varchar 10	Ano

3.2.13 Tabulka délka docházky

Tabulka eviduje datum, kdy dítě navštívilo dětskou skupinu a dále čas strávený v dětské skupině, aby bylo možno účtovat rodičům docházku jejich dítěte.

Tabulka č. 18: Tabulka délka docházky (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_delka	Identifikační číslo délky docházky	Int	Ano
Datum	Datum kdy dítě bylo v dětské skupině	Date	Ano
Čas	Strávený čas v dětské skupině	Time	Ano

3.2.14 Tabulka Přihláška

Tabulka eviduje všechny údaje, které se vyplňují v přihlášce. Od rodičů dětí, co navštěvují dětskou skupinu se zjistí například kdo mimo rodinu má dítě z dětské skupiny vyzvedávat, jestli bude dítě chodit na placené obědy, alergie a týdenní docházka.

Tabulka č. 19: Tabulka přihláška (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_prihlaska	Identifikační číslo přihlášky	Int	Ano
Jmeno	Jméno klienta	Varchar 30	Ano
Prijmeni	Příjmení klienta	Varchar 40	Ano
RC	Rodné číslo klienta	Varchar 11	Ano
Datum_narozeni	Datum narození klienta	Date	Ano
Jmeno_otce	Jméno otce dítěte	Varchar 30	Ano

Prijmeni_otce	Příjmení otce dítěte	Varchar 40	Ano
Kontakt_otce	Kontakt otce dítěte	Varchar 16	Ano
Kontakt_matky	Kontakt matky dítěte	Varchar 16	Ano
Prijmeni_matky	Příjmení matky dítěte	Varchar 40	Ano
Jmeno_matky	Jméno matky dítěte	Varchar 30	Ano
Prijmeni_otce	PSČ bydliště klienta	Int	Ano
Kontakt_otce	Přiřazení přihlášky klientovi	Int	Ne
Vyzvedavani	Eviduje jména, které můžou dítě odevzdávat	Int	Ne
Alergie	Určuje, zda má klient alergie, případně jaké	Int	Ano
Rodny_list	Eviduje, zda rodiče odevzdaly rodný list dítěte	Int	Ano
Ockovaci_prukaz	Eviduje, zda rodiče odevzdaly očkovací průkaz dítěte	Int	Ano
Obedy	Určuje, zda dítě bude chodit na placené obědy v dětské skupině	Int	Ano
Tydeni_dochazka	Určuje, předpokládanou týdenní docházku dítěte	Int	Ano
Pobočka	Určuje, na jakou pobočku si dal klient přihlášku	Int	Ano

3.2.15 Tabulka vyzvedávání

Tabulka vyzvedávání eviduje jméno a příjmení osoby, která může dítě z dětské skupiny vyzvedávat.

Tabulka č. 20: Tabulka vyzvedávání (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_vyzvedavani	Identifikační číslo vyzvedávání dítěte	Int	Ano
Jmeno	Jméno osoby, která může dítě vyzvedávat z dětské skupiny	Varchar 35	Ano
Prijmeni	Příjmení osoby, která může dítě vyzvedávat z dětské skupiny	Varchar 40	Ano

3.2.16 Tabulka alergie

Tabulka alergie eviduje, zda dítě trpí alergiemi.

Tabulka č. 21: Tabulka alergie (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_alergie	Identifikační číslo alergií dítěte	Int	Ano
Popis	Název alergie	Varchar 20	Ano

3.2.17 Tabulka rodný list dítěte

Tabulka eviduje, zda rodiče společně s přihláškou přidali kopii rodného listu dítěte.

Tabulka č. 22: Tabulka rodný list dítěte (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_rodny	Identifikační číslo rodného listu dítěte	Int	Ano
Odevzdani	Zda rodiče odevzdaly rodný list dítěte	Boolean	Ano

3.2.18 Tabulka očkovací průkaz

Tabulka eviduje, zda rodiče společně s přihláškou přidali kopii očkovacího průkazu dítěte. Bez něj by dítě nemohlo navštěvovat dětskou skupinu.

Tabulka č. 23: Tabulka očkovací průkaz (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_ockovaci	Identifikační číslo očkovacího průkazu dítěte	Int	Ano
Odevzdani	Zda rodiče odevzdaly očkovací průkaz dítěte	Boolean	Ano

3.2.19 Tabulka týdenní docházky

Tabulka eviduje dny v týdnu, ve kterých může majitel dětské skupiny počítat s příchodem do dětské skupiny. Tato tabulka pomáhá majitelům s rozložením zaměstnanců, tak aby zvládali práci s dětmi.

Tabulka č. 24: Tabulka týdenní docházka (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_dochazka	Identifikační číslo týdenní docházky dítěte	Int	Ano
Dny	Ve, které dny bude dítě pravidelně chodit do dětské skupiny	Text	Ano

3.2.20 Tabulka obědy

Tabulka eviduje, zda bude dítě chodit na obědy v dětské skupině, nebo jestli si bude nosit jídlo vlastní, které se bude ohřívat.

Tabulka č. 25: Tabulka obědy (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_obed	Identifikační číslo obědu dítěte	Int	Ano
Placene_obedy	Zda bude pravidelně chodit dítě v dětské skupině na oběd	Boolean	Ano

3.2.21 Dekompoziční tabulka počet docházky

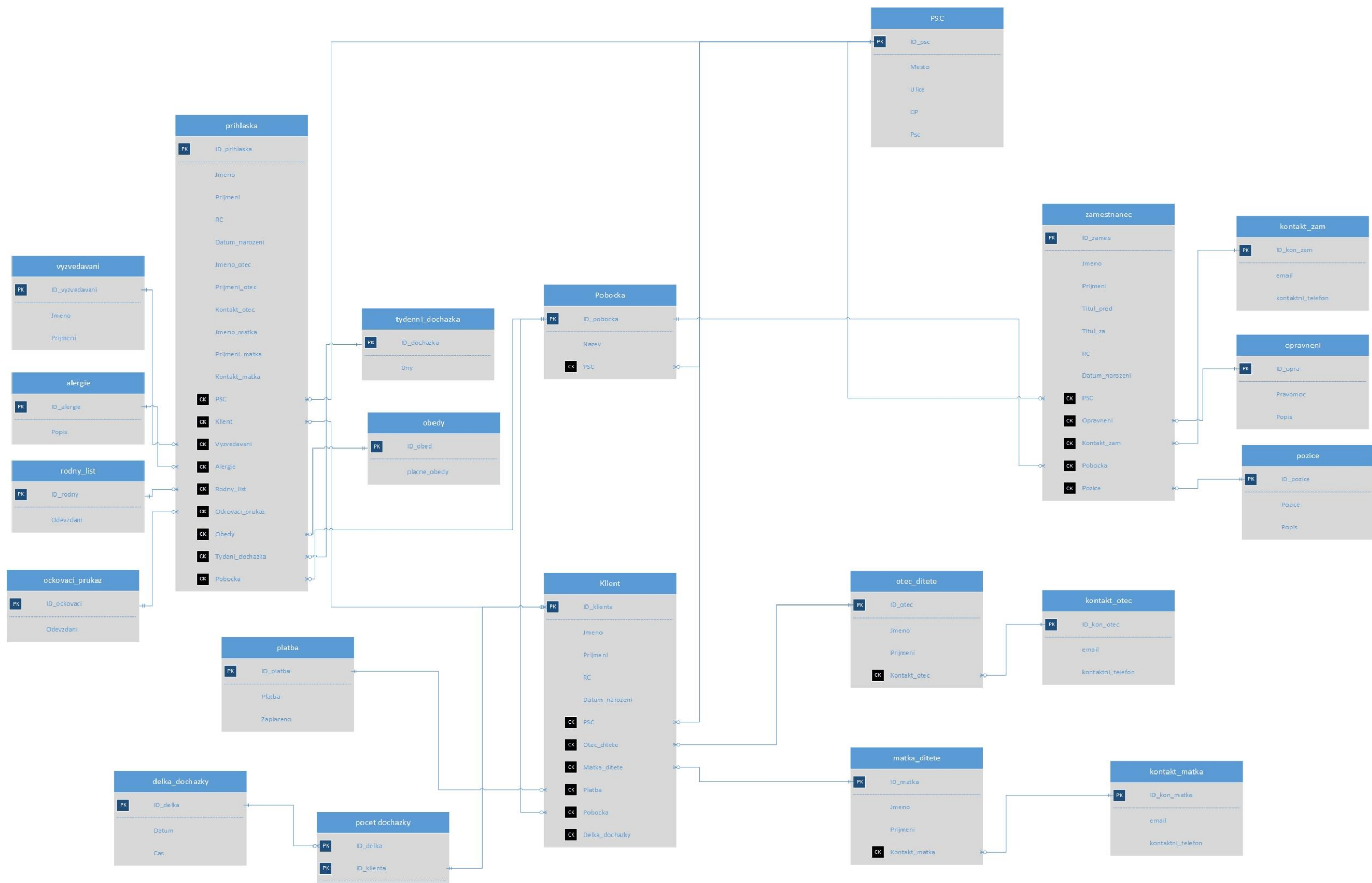
Dekompoziční tabulka nám slouží pro dekompozice tabulek klienta a délky docházky. Aby bylo možné dát jednomu klientovi více docházkových dnů.

Tabulka č. 26: Dekompoziční tabulka počet docházky (Zdroj: Vlastní zpracování)

Název atributu	Popis atributu	Typ a délka	NOT NULL
ID_delka	Identifikační číslo délky docházky dítěte	Int	Ano
ID_klienta	Identifikační číslo klienta	Int	Ano

3.2.22 E-R diagram

Obrázek č. 22 zobrazuje ER – diagram navrhnuté databáze dětské skupiny. V diagramu jsou zaznačeny všechny tabulky s jejich relacemi. Každá tabulka obsahuje název a všechny atributy, které jsou zobrazeny bez jejich datových typů, ty jsou k zobrazení v tabulkách ve třetím sloupci v předchozích kapitolách. Každá tabulka v E-R diagramu obsahuje název primárního klíče, který je znázorněn pomocí modrého čtverce s písmeny PK. Pokud tabulka obsahuje cizí klíče, tak ty jsou znázorněny pomocí černého čtverce s písmeny CK.



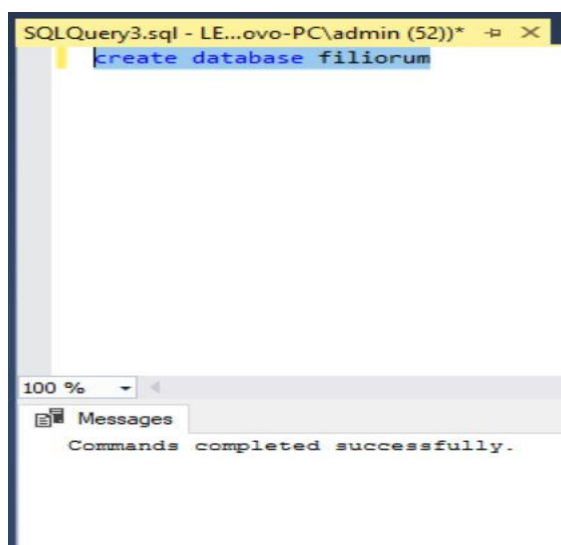
Obrázek č. 22: E-R diagram (Zdroj: Vlastní zpracování)

3.1 Fyzický návrh

Fyzický návrh databáze se vytváří na základě logického a E-R diagramu, první krok je převést logický návrh databáze do požadovaného stavu. V tomto kroku se musí vytvořit návrh předpokládaných tabulek společně s návrhem reprezentujícím odvození dat a vytvořit návrh integritních omezení. V druhém kroku se musí vytvořit organizace souborů a indexů. Druhý krok bude obsahovat analýzu transakcí, volbu organizačních souborů a volbu indexů. Ve třetím kroku se navrhnu pohledy. Ve čtvrtém kroku se musí zavést redundance. Poslední krok je monitorování a vyladění systému v provozu.

3.1.1 Vytvoření databáze

Před vytvořením tabulek musíme vytvořit vlastní databázi. Databáze musí mít vlastní jméno, v našem případě jsem si databázi pojmenoval po jménu dětské skupiny. Databáze se vytváří pomocí příkazu CREATE DATABASE Filiorum. **Obrázek č. 23** znázorňuje vytvoření databáze.



Obrázek č. 23: Vytvoření databáze (Zdroj: Vlastní zpracování)

3.1.2 Vytvoření tabulek

V této kapitole vytvoříme tabulky, které jsme si nadefinovali v logickém a fyzickém návrhu. Tabulky vytváříme v příslušné databázi. Tabulky vytváříme pomocí příkazu CREATE TABLE. Každý sloupec v příkazu obsahuje název sloupce, datový typ sloupce a případně omezení sloupce. Omezení sloupce chápeme primární klíče, nulové a nenulové atributy.

Z E-R diagramu je zřetelné, že se musí nejdříve vytvořit „hlavní“ tabulky a po nich číselníky a zbytek tabulek. Hlavními tabulkami jsou tabulka zaměstnanců, tabulka přihlášky a tabulka klientů, kdyby měla dětská skupina více poboček, tak by mezi hlavní tabulku patřily i pobočky. V této kapitole jsem zobrazil vytvoření tabulky zaměstnanci, která má atributy z kapitoly 3.2. Vytvoření všech zbylých tabulek lze nalézt v příloze.

Zaměstnanec

```
create table zamestnanec (  
ID_zames int not null identity (1,1) primary key,  
jmeno varchar (35) not null,  
prijmeni varchar (40) not null,  
titul_pred varchar (5),  
titul_zatva varchar (5),  
RC varchar (11)not null,  
datum_narozeni date not null,  
psc int not null,  
pozice int not null,  
opraveni int not null,  
kontakt_zam int not null,  
pobocka int not null,  
);
```

3.1.3 Spojení tabulek pomocí klíčů

V kapitole 3.2 jsem si nadefinoval, jaký atributy budou primárními a cizími klíči. Máme dva způsoby vytvoření vazeb mezi tabulkami, první je vytvořit vazby zároveň s vytvořením tabulek a druhý způsob, který jsem použil je vytvoření vazeb, až po vytvoření příslušných tabulek.

Ukážeme si vytvoření vazeb z tabulky zaměstnance. Z E-R diagramu víme, že tabulka zaměstnance má celkem pět cizích klíčů, tím pádem dohromady povede z tabulky zaměstnance pět vazeb, které se musí spojit s primárními klíči daných tabulek. K tomu slouží příkaz (tzv. foreign key - FK). Script obsahuje vytvoření vazeb tabulky zaměstnance, zbytek vazeb můžeme najít v příloze.

Vazby tabulky zaměstnanec

```
alter table zamestnanec add  
foreign key (psc) references psc(ID_psc),  
foreign key (kontakt_zam) references kontakt_zam(ID_konm),  
foreign key (pozice) references pozice(ID_pozice),  
foreign key (opraveni) references opraveni(ID_opra),  
foreign key (pobocka) references pobocka(ID_pobocka);
```

3.1.4 Naplnění tabulky daty

Abych neohrozil soukromí klientů dětské skupiny zveřejněným jejich osobních údajů, tak jsem k naplnění tabulek použil fiktivní data, která nemají s dětskou skupinou nic společného a nejedná se ani o osobní údaje reálných osob.

Do relace lze data zadávat pomocí příkazu INSERT jazyka SQL. Naplnění tabulek lze provést více způsoby. V prvním způsobu si můžeme vybrat atributy, který chceme vyplnit, v tom případě ale k příkazu musíme napsat atributy o které se jedná. Ve druhém způsobu zadáváme ke každému atributu data, tento způsob jsem si vybral ve své práci. A ve třetím způsobu můžeme data hromadně vložit z jiné tabulky. Scripty jednotlivých způsobu vložení zaměstnance jsou pod textem. Zbytek scriptů vložení dat do databáze lze nalézt v příloze.

Varianta 1

```
insert into zamestnanec (jmeno, prijmeni, titul_pred, titul_z, RC, datum_nar) values  
('Jana', 'Pitrochová', "", "", '963007/4231', '1996-07-30');
```

Varianta 2

```
insert into zamestnanec values  
('Jana', 'Pitrochová', "", "", '963007/4231', '1996-07-30', 1, 2, 1, 1, 1),  
('Honza', 'Troška', "", "", '903007/4231', '1990-07-30', 2, 4, 2, 2, 1),  
('Zuzana', 'Vlná', 'Ing', 'Dis', '765623/9876', '1976-06-23', 3, 6, 1, 3, 1);
```

Varianta 3

```
insert into zamestnanec  
select * from zamestnanec 2  
where id > 1
```

Další způsob naplnění tabulky daty je přímo při rozkliknutí tabulky v databázi Filiorum, kde můžeme ručně přidat, či upravit data. **Obrázek č. 24** zobrazuje způsob přidání dat do tabulky zaměstnanců, kdy do políček Null lze vyplnit vlastní data.

ID_zames	jmeno	prijmeni	titul_pred	titul_zs	RC	datum_narozeni	psc
5	Jana	Pitrochová			963007/4231	1996-07-30	1
18	Honza	Troška	NULL	NULL	903007/4231	1990-07-30	2
19	Zuzana	Volná	Ing	Dis	765623/9876	1976-06-23	3
20	Honza	Mitricha		Dis	903007/4231	1990-07-30	2
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Obrázek č. 24: Přidání dat (Zdroj: Vlastní zpracování)

3.2 Pohledy

Tato kapitola popisuje pohledy, které pomohou majitelům s lepší správou dětské skupiny. Jedná se o virtuální tabulku, která je vytvořená pomocí výsledného dotazu. S pohledem neboli VIEW můžeme pracovat jako se skutečnými tabulkami. V jednotlivých podkapitolách se budu zabývat pohledy, které se budou používat v dětské skupině.

3.2.1 Pohled klientů na jednotlivých pobočkách

Pomocí tohoto pohledu mohou majitelé a případně správcové systému zobrazit jednotlivé klienty, kteří navštěvují dané pobočky dětské skupiny. Vzhledem k tomu, že v současné době má dětská skupina jen jednu pobočku je tento pohled nepotřebný, ale je nachystaný do budoucna, pokud se dětská skupina rozroste. Z tohoto důvodu jsem si do tabulky klientů vložil klienta s druhou pobočkou, abych viděl, zda skript funguje. Tento pohled zobrazuje jména a příjmení klientů a pobočku, kterou navštěvují. Níže je k vidění skript pohledu společně se skriptem vložení klienta s jinou pobočkou. Informace z pohledu zobrazuje **obrázek č. 25**.

```
insert into klient values
```

```
('Vojta','Tichý','153007/4231','2015-07-30',1,1,2,1,1);
```

```
create view deti_pobocka as
```

```
select kl.jmeno + ' ' + kl.prijmeni as 'klient', po.nazev as 'Pobočka'
```

```
from klient kl left join pobocka po on kl.pobocka = po.ID_pobocka
```

```
go
```

	klient	Pobočka
1	Vojta Novák	Zvoneček
2	Honza Stálý	Zvoneček
3	Dana Stálá	Zvoneček
4	Honza Voda	Zvoneček
5	Jana Kolářová	Zvoneček
6	Zuzana Dobešová	Zvoneček
7	Vášek Hot	Zvoneček
8	Vojta Tichý	Hříbeček

Obrázek č. 25: Pohled klienti na pobočkách (Zdroj: Vlastní zpracování)

Pokud budou chtít majitelé dětské skupiny zobrazit jen klienty z jedné pobočky, tak se použije skript s podmínkou na jméno pobočky. Níže je k vidění script společně s **obrázkem č. 26**, který zobrazuje upravený pohled.

```
create view deti_pobocka_2 as
select kl.jmeno + ' ' + kl.prijmeni as 'klient', po.nazev as 'Pobočka'
from klient kl left join pobocka po
on kl.pobocka = po.ID_pobocka where kl.pobocka = 1
go
```

	klient	Pobočka
1	Vojta Novák	Zvoneček
2	Honza Stálý	Zvoneček
3	Dana Stálá	Zvoneček
4	Honza Voda	Zvoneček
5	Jana Kolářová	Zvoneček
6	Zuzana Dobešová	Zvoneček
7	Vášek Hot	Zvoneček

Obrázek č. 26: Klienti na pobočce Zvoneček (Zdroj: Vlastní zpracování)

3.2.2 Pohled Zaměstnanců na pobočkách

Tento pohled slouží stejně jako předchozí k budoucím účelům, kdy si budou moci majitelé zobrazit svoje zaměstnance, kteří pracují na jednotlivých pobočkách. I v tomto případě se může daný skript upravit o podmínku, která bude určovat, jaký zaměstnanci na daných pobočkách se zobrazí při spuštění pohledu. Pohled obsahuje jména a příjmení zaměstnanců i s jejich tituly, dále zobrazuje jméno pobočky, na které zaměstnanec pracuje. Skript i s **obrázkem č. 27** je pod textem.

```
create view zamestnanci_pobocka_2 as
select za.titul_pred as 'Titul před jménem', za.jmeno + ' ' + za.prijmeni as 'zaměstnanec',
za.titul_zo as 'Titul za jménem', replace(replace(za.pobocka,1,'Zvoneček'),2,'Hříbeček')
as 'Pobočka'
from zamestnanec za left join pobocka po on za.ID_zames=po.ID_pobocka
go
```

	Titul před jménem	zaměstnanec	Titul za jménem	Pobočka
1		Jana Pitrochová		Zvoneček
2	NULL	Honza Troška	NULL	Zvoneček
3	Ing	Zuzana Volná	Dis	Zvoneček
4		Honza Mitricha	Dis	Hříbeček

Obrázek č. 27: Zaměstnanci na pobočkách (Zdroj: Vlastní zpracování)

3.2.3 Pohled oprávnění zaměstnanců

Tento pohled slouží k zobrazení všech zaměstnanců pracujících pro dětskou skupinu, tento pohled slouží hlavně k zobrazení pravomoci a oprávnění každého zaměstnance. Pohled kromě jména a příjmení zobrazuje zmiňované oprávnění a pravomoci a dále osobní informace a kontaktu zaměstnance. **Obrázek č. 28** a skript níže zobrazuje pohled oprávnění zaměstnanců.

```
create view zamestnanci_pozice_opravneni as
select za.titul_pred as 'Titul před jménem', za.jmeno + ' ' + za.prijmeni as 'zaměstnanec',
za.titul_zo as 'Titul za jménem', za.RC , za.datum_narozeni as 'Datum narození',
po.pozice as 'Pozice',
op.pravomoc as 'Oprávnění', ko.kontaktni_telefon as 'Telefon'
from zamestnanec za left join opravneni op on za.opravneni=op.ID_opra
left join pozice po on za.pozice=po.ID_pozice
left join kontakt_zam ko on za.kontakt_zam=ko.ID_konm
go
```

	Titul před jménem	zaměstnanec	Titul za jménem	RC	Datum narození	Pozice	Oprávnění	Telefon
1		Jana Pítrchová		963007/4231	1996-07-30	recepční	vkładani	763 345 007
2	NULL	Honza Troška	NULL	903007/4231	1990-07-30	uklizečka	bez prav	753 229 876
3	Ing	Zuzana Volná	Dis	765623/9876	1976-06-23	ředitelka	admin	780 150 007
4		Honza Mitricha	Dis	903007/4231	1990-07-30	správce	bez prav	753 229 876

Obrázek č. 28: Pravomoc zaměstnanců (Zdroj: Vlastní zpracování)

3.2.4 Docházka klientů

Pomocí pohledu docházka klientů si mohou majitelé zobrazit docházku jednotlivých klientů v určitém období. Jelikož majitelé každý měsíc posílají rodičům dětí v dětské skupině fakturu, kolik mají zaplatit, tak se jim tento pohled hodí. Pohled zobrazuje jméno a příjmení klienta společně s datumem docházky a časem (hh:mm:ss) stráveným v dětské skupině.

Díky tomuto pohledu si mohou majitelé zobrazit docházku na daný měsíc a propočítat kolik mají rodiče zaplatit za docházku. Posléze správce změní informace o zaplacení u daného klienta z nuly na jedničku, aby bylo možné poznat kdo docházku zaplatil.

Obrázek č. 29 společně se skriptem zobrazují pohled docházky všech klientů ve všech obdobích.

```
create view dochazka_klientu as
select kl.jmeno + ' '+kl.prijmeni as 'Klient', de.datum as 'Datum',
de.cas as 'Čas'
from klient kl join pocet_dochazky po on kl.ID_klienta=po.ID_klienta
left join delka_dochazky de on de.ID_delka=po.ID_delka
go
```

	Klient	Datum	Čas
1	Vojta Novák	2019-01-01	08:00:00.
2	Vojta Novák	2019-01-02	05:00:00.
3	Vojta Novák	2019-01-04	05:00:00.
4	Dana Stálá	2019-01-01	08:00:00.
5	Dana Stálá	2019-01-02	05:00:00.
6	Dana Stálá	2019-01-03	08:00:00.
7	Honza Voda	2019-01-01	08:00:00.
8	Honza Voda	2019-01-04	05:00:00.
9	Jana Kolářová	2019-01-01	06:30:00.
10	Jana Kolářová	2019-01-04	05:00:00.

Obrázek č. 29: Docházka klientů (Zdroj: Vlastní zpracování)

Skript níže společně s **obrázkem č. 30** zobrazuje docházku klientů k určitému datu. Pohled zobrazuje jméno a příjmení klientů společně s docházkou a časem (hh:mm:ss) strávený v dětské skupině k určitému datu.

```
create view dochazka_klientu_2 as
select kl.jmeno + ' '+kl.prijmeni as 'Klient', de.datum as 'Datum', de. cas as 'Čas'
from klient kl join pocet_dochazky po on kl.ID_klienta=po.ID_klienta
left join delka_dochazky de on de.ID_delka=po.ID_delka
where de.datum = '2019-01-01'
go
```

	Klient	Datum	Čas
1	Vojta Novák	2019-01-01	08:00:00.
2	Dana Stálá	2019-01-01	08:00:00.
3	Honza Voda	2019-01-01	08:00:00.
4	Jana Kolářová	2019-01-01	06:30:00.
5	Zuzana Dobešová	2019-01-01	06:30:00.
6	Vašek Hot	2019-01-01	05:00:00.

Obrázek č. 30: Docházka klientů k určitému datu (Zdroj: Vlastní zpracování)

3.2.5 Pohled přihláška

Pohled přihláška slouží k zobrazení, jestli rodiče klienta odevzdali očkovací průkaz a rodný list dítěte společně s přihláškou. Pohled zobrazuje jméno a příjmení klienta společně s jeho rodiči, dále zobrazuje, kdo dítě kromě rodiny může vyzvedávat, alergie, a jestli má dítě obědy vlastní nebo se bude oběd kupovat v dětské skupině. Pohled společně se skriptem zobrazuje **Obrázek č. 31**.

```
create view prihlaska_zobrazeni as
select pr.jmeno + ' '+pr.prijmeni as 'Klient', pr.jmeno_mamky + ' '+pr.prijmeni_mamky
as 'Mamka',
pr.jmeno_otce + ' '+pr.prijmeni_otce as 'Otec', vy.jmeno + ' '+vy.prijmeni as
'Vyzvedavani',
al.popis 'Alergie', replace(replace(ro.odevzdani,1,'Odevzdal'),2,'Neodevzdal') as
'Rodný list',
replace(replace(oc.odevzdani,1,'Odevzdal'),2,'Neodevzdal') as 'Očkovací průkaz',
replace(replace(ob.placene,1,'Kupované'),2,'Vlastní obědy') as 'Obědy'
from prihlaska pr left join vyzvedavani vy on pr.vyzvedavani=vy.ID_vyzvedavani
left join alergie al on pr.alergie = al.ID_alergie
left join rodny_list ro on pr.rodny_list=ro.ID_rodny
```

```

left join ockovaci_prukaz oc on pr.ockovaci_prukaz=oc.ID_ockovaci
left join obedy ob on pr.obedy=ob.ID_obed
go

```

	Klient	Mamka	Otec	Vyzvedavani	Alergie	Rodný list	Očkovací průkaz	Obědy
1	Vojta Novák	Hana Novákáková	Jan Novák	Hana Novakova	ořechy	Odevzdal	Odevzdal	Kupované
2	Honza Stálý	Tereza Stálá	Jidáš Stálý	Zuzana Ticha	bodnutí	Odevzdal	Odevzdal	Kupované

Obrázek č. 31: Přihláška klienta (Zdroj: Vlastní zpracování)

3.3 Snapshot

Jelikož návrh databáze je vytvořen v SQL, a ten poskytuje možnost rychlé zálohy před změnami, které by mohli ohrozit strukturu dat nebo databáze. Snapshot může v našem případě sloužit i k porovnání dat z různých časových období. Majitelům bych doporučil, aby se snapshot vytvářel každý den, aby bylo zamezeno ztráty dat za větší časové období v případě narušení struktury databáze.

Snapshot se vytváří podobně jako databáze, tedy pomocí příkazu CREATE DATABASE, ke Skriptu se připsí jméno (filegroup) a dále cesta, kam se snapshot uloží. Skript níže zobrazuje vytvoření snapshotu.

```

CREATE DATABASE db_snapshot on
(
    name = 'AdventureWorks2014_Data',
    filename = 'C:\Program Files\Microsoft SQL Server\MSAS14.SQLEXPRESS
              MSSQL\DATA\test_db_snapshot.ds'
)
As snapshot of test_d

```

K obnovení databáze nám stačí napsat skript níže do master databáze v sql, musíme ale musíme dbát na omezení popsané v **kapitole 1.9.6**.

```

Restore database test_db from database_snapshot = 'db_snapshot'

```

3.4 Trigger smazání zaměstnance

Trigger smazání zaměstnance má usnadnit práci se samotným systémem, a to v případě, kdy dojde ke smazání zaměstnance z důvodů vyhození nebo z důvodů výmazu osobních údajů o které požádá zaměstnanec.

Trigger smaže všechny údaje, které přímo souvisí s daným zaměstnancem, v našem příkladu dojde ke smazání údajů z tabulek kontakt zaměstnance a psč.

Pokud chceme pomocí triggeru smazat zaměstnance jehož ID je 5, tak použijeme následující skript:

```
create trigger smazani_zamestnanec on zamestnanec
for delete as
begin
declare @id_zamestnanec int;
declare @psc int;
declare @kontakt_zamestnanec int;
select @id_zamestnanec=id_zames from deleted
set @psc=(select top 5 psc from psc where id_zames=id_zamestnanec);
set @kontakt_zam=(select top 5 kontakt_zam where zamestnanec=id_zamestnanec);
delete from zamestnanec where id_zames=@id_zamesstnanec;
delete from psc where id_psc=@id_psc;
delete from kontakt_zam where id_konm=@id_kontakt_zamestnanec;
end
go
```

Pokud chceme smazat údaje o zaměstnanci jehož ID je 5, tak použijeme skript níže, díky němuž se spustí trigger smazani_zamestnanec nad tabulkou zaměstnanci.

```
delete from zamestnanec where id_zames = 5
```

3.5 Zhodnocení návrhu

Největším přínosem vytvořeného návrhu je řešení nedostatků vyplývajících z kapitoly 2.8. Díky tomuto návrhu bude dětská skupina moci lépe a přehledněji evidovat svá data a zároveň bude mít i lépe zvládnuté zálohování dat, které je v současnosti řešeno jen v části případů.

Aby byl výše popsáný návrh databázového systému přijatelný pro koncové uživatele je ho ještě nutné doplnit o uživatelské aplikační rozhraní, což však není předmětem této práce.

Ekonomické přínosy návrhu není momentálně možné přesně určit, vzhledem k tomu, že implementace popsaného návrhu nebyla prozatím provedena, a tudíž nemohly nastat přínosy s touto implementací spojené. Nicméně můžeme předpokládat, že přínosy pro dětskou skupinu budou převažovat náklady na tento systém.

ZÁVĚR

Hlavním cílem této práce bylo vytvořit databázový systém pro dětskou skupinu, působící v Brně.

Databázový systém byl navržen po konzultaci s vedením dětské skupiny, tak aby usnadnil majitelům správu dětské skupiny. Systém byl navržen, tak aby vyhovoval i budoucím požadavkům s přidáním další pobočky dětské skupiny. Databázový systém je možno dále rozšířit nebo upravit.

Dětská skupina neměla jednotný systém, který by umožňoval rychlé vyhledávání informací ohledně klientů nebo zaměstnanců. Chyběla jakákoliv záloha dat, která by zabránila poškození nebo zničení jednotlivých formulářů nebo paměťového média v počítači. Při práci s docházkou dětí museli majitelé projít nepřehledné formuláře a získat z nich potřebné informace, které potřebovali.

Navržený systém umožňuje evidovat informace o klientech, zaměstnancích a zároveň umožňuje jednotlivé propojení mezi nimi a jednotlivými pobočkami a ostatními tabulkami. Velká výhoda databázového systému je v přehlednosti dat a v možnosti rychlého vyhledání potřebných informací. Databázový systém umožňuje zálohu a analýzu dat, díky které se sníží riziko nenávratné ztráty informací. Hlavní výhoda systému je v přehlednosti, která vrátí majitelům spoustu času, který mohou dále využít pro zlepšování kvalit dětské skupiny.

Navržený systém obsahuje pohledy, které usnadní majitelům nebo správcům systému práci s danými informacemi. Díky pohledům vzroste přehlednost, díky které budou mít lepší náhled na celý systém.

Tato práce v první kapitole shrnuje problém s osobními údaji podle nařízení od evropské unie, práce poskytuje správcům systému informace o právech a povinnostech při práci s osobními údaji.

SEZNAM POUŽITÝCH ZDROJŮ

- [1] KOCH, Miloš a Viktor ONDRÁK. *Informační systémy a technologie*. Vyd. 3. Brno: Akademické nakladatelství CERM, 2008. ISBN isbn978-80-214-3732-6.
- [2] KOCH, Miloš a Bernard NEUWIRTH. *Datové a funkční modelování*. Vyd. 4., rozš. Brno: Akademické nakladatelství CERM, 2010. ISBN isbm978-80-214-4125-5.
- [3] KROENKE, David a David J. AUER. *Databáze*. Brno: Computer Press, 2015. ISBN isbn978-80-251-4352-0.
- [4] CONOLLY, Thomas, Carolyn E. BEGG a Richard HOLOWCZAK. *Mistrovství - databáze: profesionální průvodce tvorbou efektivních databází*. Brno: Computer Press, 2009. ISBN isbn978-80-251-2328-7.
- [5] mvcr.cz: co-je-gdpr [online]. Ministerstvo vnitra České republiky, ©2019 [cit. 2019-05-06]. Dostupné z: <https://www.mvcr.cz/gdpr/clanek/co-je-gdpr.aspx>
- [6] MV ČR. *Stručný popis obsahu nového Obecného nařízení o ochraně osobních údajů* [online]. 2016, 15 [cit. 2019-05-07]. Dostupné z: <https://www.mvcr.cz/gdpr/soubor/gdpr-sesit-pdf.aspx>
- [7] Uoou: *GDPR (obecné nařízení)* [online]. Jihlava: Úřad pro ochranu osobních údajů, ©2013 [cit. 2019-05-06]. Dostupné z: <https://www.uoou.cz/gdpr/ds-3938/p1=3938>
- [8] Interval.cz: *SQL – tvorba tabulek* [online]. ZONER software, a.s, 2015 [cit. 2019-05-06]. Dostupné z: <https://www.interval.cz/clanky/sql-tvorba-tabulek/>
- [9] KOUTSKÝ, Tonda. *Koutsky.info: SQL* [online]. Joomla Templates, 2015 [cit. 2019-05-06]. Dostupné z: <http://koutsky.info/index.php/10-sql/8-sql-vkladani-zaznamu-do-tabulky>
- [10] DVOŘÁK, Michal. *Wordpress: database* [online]. WordPress, 2010 [cit. 2019-05-06]. Dostupné z: <https://midv.wordpress.com/2009/03/07/database-snapshot-zaloha-na-pokusy-za-par-sekund/>
- [11] FEKETE, Alan, Dimitrios LIAROKAPIS, Elizabeth O'NEIL, Patrick O'NEIL a Dennis SHASHA. Making snapshot isolation serializable. *ACM Transactions on Database Systems (TODS)* [online]. ACM, 2005, 30(2), 492-528 [cit. 2019-05-04]. DOI: 10.1145/1071610.1071615.

- [12] BAČINA, Ondřej. Snapshoty. *My-ty: Snapshoty* [online]. my-ty, ©2008-2019 [cit. 2019-05-06]. Dostupné z: <https://www.optimalizovane-it.cz/storage/snapshots-zpusob-jak-posunout-zalohovani-a-obnovu-dale.html>
- [13] BELGIE. *NAŘÍZENÍ EVROPSKÉHO PARLAMENTU A RADY (EU): o ochraně fyzických osob*. In: . Brusel: Evropský parlament, 2016, ročník 2016, 2016/679.
- [14] Viznerová, J. *Organizační struktura podniku* [ústní sdělení]. Brno 13.2.2019.
- [15] Viznerová, J. *Žádost o přijetí dítěte*. Brno 13.2.2019.
- [16] Viznerová, J. *Zmocnění osob klientů DS*. Brno 13.2.2019.
- [17] Viznerová, J. *Vstupní dotazník klientů DS*. Brno 13.2.2019.
- [18] OPPEL, Andrew J. *SQL bez předchozích znalostí: [průvodce pro samouky]*. Brno: Computer Press, 2008. ISBN isbn978-80-251-1707-1.

SEZNAM OBRÁZKŮ

Obrázek č. 1: Znázornění informací.....	14
Obrázek č. 2: Znázornění dat.....	15
Obrázek č. 3: Komponenty databázového systému	16
Obrázek č. 4: Lineární datový model	21
Obrázek č. 5: Objektový datový model	22
Obrázek č. 6: Hierarchický datový model	22
Obrázek č. 7: Síťový model.....	23
Obrázek č. 8: Relační datový model	24
Obrázek č. 9: Vztah 1:1	27
Obrázek č. 10: Vztah 1:N	27
Obrázek č. 11: Vztah N:M.....	28
Obrázek č. 12: Organizační struktura firmy.....	37
Obrázek č. 13: Evidence klienta	38
Obrázek č. 14: Zmocnění osob	39
Obrázek č. 15: Informace o dítěti.....	40
Obrázek č. 16: Evidence klientů diagram.....	41
Obrázek č. 17: Tabulka o současných klientech	42
Obrázek č. 18: Tabulka o budoucích klientech	42
Obrázek č. 19: Tabulka o minulých klientech.....	43
Obrázek č. 20: Evidence docházky	44
Obrázek č. 21: Identifikace relací	48
Obrázek č. 22: E-R diagram	61
Obrázek č. 23: Vytvoření databáze	62
Obrázek č. 24: Přidání dat	65
Obrázek č. 25: Pohled klienti na pobočkách.....	66
Obrázek č. 26: Klienti na pobočce Zvoneček	66
Obrázek č. 27: Zaměstnanci na pobočkách.....	67

Obrázek č. 28: Právní moc zaměstnanců	68
Obrázek č. 29: Docházka klientů	68
Obrázek č. 30: Docházka klientů k určitému datu.....	69
Obrázek č. 31: Přihláška klienta	70

SEZNAM TABULEK

Tabulka č. 1: Obsah databáze	17
Tabulka č. 2: Systémy řízení databáze	17
Tabulka č. 3: Funkce databázových aplikací	20
Tabulka č. 4: Vlastnosti relací	25
Tabulka č. 5: identifikace entit	46
Tabulka č. 6: Tabulka zaměstnanec	49
Tabulka č. 7: Tabulka PSC.....	50
Tabulka č. 8: Tabulka pozice.....	51
Tabulka č. 9: Tabulka zaměstnanec	51
Tabulka č. 10: Tabulka oprávnění.....	51
Tabulka č. 11: Tabulka pobočka	52
Tabulka č. 12: Tabulka klient	52
Tabulka č. 13: Tabulka otec dítěte	53
Tabulka č. 14: Tabulka kontakt otec dítěte	54
Tabulka č. 15: Tabulka matka dítěte	54
Tabulka č. 16: Tabulka kontakt matky dítěte	55
Tabulka č. 17: Tabulka platba.....	55
Tabulka č. 18: Tabulka délka docházky	56
Tabulka č. 19: Tabulka přihláška	56
Tabulka č. 20: Tabulka vyzvedávání.....	58
Tabulka č. 21: Tabulka alergie	58
Tabulka č. 22: Tabulka rodný list dítěte	58
Tabulka č. 23: Tabulka očkovací průkaz	59
Tabulka č. 24: Tabulka týdenní docházka	59
Tabulka č. 25: Tabulka obědy	59

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

SQL	Structured query language
GDPR	General data protection regulation
IBM	International business machines
SŘBD	Systém řízení báze dat
DDL	Data definition language
DQL	Database query language
DML	Data manipulation language
MS	Microsoft software
DS	Dětská skupina
PSC	Poštovní směrovací číslo
RC	Rodné číslo
FK	Foreign key
ID	Identifikační číslo
PK	Primary key
CP	Číslo popisné

SEZNAM PŘÍLOH

Příloha č. 1: Vytvoření tabulek

Příloha č. 2: Vytvoření klíčů k vytvořeným tabulkám

Příloha č. 3: Naplnění tabulek fiktivními daty

Příloha č. 1: Vytvoření tabulek

```
create table zamestnanec (  
ID_zames int not null identity (1,1) primary key,  
jmeno varchar (35) not null,  
prijmeni varchar (40)not null,  
titul_pred varchar (5),  
titul_zo varchar (5),  
RC varchar (11)not null,  
datum_narozeni date not null,  
psc int not null,  
pozice int not null,  
opraveni int not null,  
kontakt_zam int not null,  
pobocka int not null,  
);  
create table kontakt_zam(  
ID_konm int not null identity (1,1) primary key,  
email varchar (50) not null,  
kontaktní_telefon varchar (16) not null,  
);  
create table psc(  
ID_psc int not null identity (1,1) primary key,  
psc varchar (6) not null,  
mesto varchar (40) not null,  
ulice varchar (40) not null,  
cp varchar (3) not null,  
);  
create table pozice(  
ID_pozice int not null identity (1,1) primary key,  
pozice varchar (20) not null,  
popis text,  
);  
create table opraveni(  
ID_opra int not null identity (1,1) primary key,  
pravomoc varchar (50) not null,  
popis text,  
);
```



```

create table pobočka(
ID_pobočka int not null identity (1,1) primary key,
nazev varchar (30) not null,
psc int not null,
);
create table klient (
ID_klienta int not null identity (1,1) primary key,
jmeno varchar (35) not null,
prijmeni varchar (40)not null,
RC varchar (11)not null,
datum_nar date not null,
otec_ditete int not null,
matka_ditete int not null,
pobočka int not null,
psc int not null,
platba int not null,
);
create table matka_ditete (
ID_matka int not null identity (1,1) primary key,
jmeno varchar (35) not null,
prijmeni varchar (40)not null,
kontakt_matka int not null,
);
create table otec_ditete (
ID_otec int not null identity (1,1) primary key,
jmeno varchar (35) not null,
prijmeni varchar (40)not null,
kontakt_otec int not null,
);
create table platba(
ID_platba int not null identity (1,1) primary key,
zaplaceno bit not null,
platba varchar (12) not null,
);
Create table delka_dochazky(
ID_delka int not null identity (1,1) primary key,
datum date not null,
cas time not null,
);

```

```

create table pocet_dochazky(
ID_klienta int not null,
ID_delka int not null,
primary key (ID_klienta, ID_delka)
);
create table kontakt_matka(
ID_konm int not null identity (1,1) primary key,
email varchar (50) not null,
kontaktní_telefon varchar (16) not null,
);
create table kontakt_otec(
ID_konm int not null identity (1,1) primary key,
email varchar (50) not null,
kontaktní_telefon varchar (16) not null,
);
create table prihlaska (
ID_prihlaska int identity (1,1) primary key not null,
jmeno varchar (30) not null,
prijmeni varchar (40) not null,
RC varchar (16) not null,
datum_narozeni date not null,
jmeno_mamky varchar (30) not null,
prijmeni_mamky varchar (40) not null,
kontakt_mamka varchar (16) not null,
jmeno_otce varchar (30) not null,
prijmeni_otce varchar (40) not null,
kontakt_otce varchar (16) not null,
PSC int not null,
klient int not null,
vyzvedavani int not null,
alergie int not null,
rodny_list int not null,
ockovaci_prukaz int not null,
obedy int not null,
tydenni_dochazka int not null,
pobocka int not null,
);

```

```

create table vyzvedavani (
ID_vyzvedavani int identity (1,1) primary key not null,
jmeno varchar (30) not null,
prijmeni varchar (40) not null,
);
create table alergie (
ID_alergie int identity (1,1) primary key not null,
popis varchar (30) not null,
);
create table rodny_list (
ID_rodny int identity (1,1) primary key not null,
odevzdani bit not null,
);
create table ockovaci_prukaz (
ID_ockovaci int identity (1,1) primary key not null,
odevzdani bit not null,
);
create table tydenni_dochazka (
ID_dochazka int identity (1,1) primary key not null,
dny text not null,
);
Create table obedy (
ID_obed int identity (1,1) primary key not null,
placene bit not null,
);

```

Příloha č. 2: Vytvoření klíčů k vytvořeným tabulkám

```
alter table zamestnanec add
```

```
foreign key (psc) references psc(ID_psc),
```

```
foreign key (kontakt_zam) references kontakt_zam(ID_konm),
```

```
foreign key (pozice) references pozice(ID_pozice),
```

```
foreign key (opraveneni) references opraveneni(ID_opra),
```

```
foreign key (pobocka) references pobocka(ID_pobocka);
```

```
alter table pocet_dochazky add
```

```
foreign key (ID_klienta) references klient(ID_klienta),
```

```
foreign key (ID_delka) references delka_dochazky(ID_delka);
```

```
alter table klient add
```

```
foreign key (otec_ditete) references otec_ditete(ID_otec),
```

```
foreign key (matka_ditete) references matka_ditete(ID_matka),
```

```
foreign key (pobocka) references pobocka (ID_pobocka),
```

```
foreign key (psc) references psc (ID_psc),
```

```
foreign key (platba) references platba (ID_platba);
```

```
alter table matka_ditete add
```

```
foreign key (kontakt_matka) references kontakt_matka(ID_konm);
```

```
alter table otec_ditete add
```

```
foreign key (kontakt_otec) references kontakt_otec(ID_konm);
```

```
alter table prihlaska add
```

```
foreign key (psc) references psc (ID_psc);
```

```
alter table prihlaska add
```

```
foreign key (klient) references klient (ID_klienta);
```

```
alter table prihlaska add
```

```
foreign key (vyzvedavani) references vyzvedavani (ID_vyzvedavani);
```

```
alter table prihlaska add
```

```
foreign key (alergie) references alergie (ID_alergie);
```

```
alter table prihlaska add
```

```
foreign key (rodny_list) references rodny_list (ID_rodny);
```

```
alter table prihlaska add
```

```
foreign key (ockovaci_prukaz) references ockovaci_prukaz (ID_ockovaci);
```

```
alter table prihlaska add
```

```
foreign key (obedy) references obedy (ID_obed);
```

```
alter table prihlaska add  
foreign key (tydenni_dochazka) references tydenni_dochazka (ID_dochazka);  
alter table prihlaska add  
foreign key (pobocka) references pobocka (ID_pobocka);
```

Příloha č. 3: Naplnění tabulek fiktivními daty

`insert into kontakt_matka values`

```
('novakova@seznam.cz', '756 230 333'),  
( 'novotna@gmail.com', '658 309 876'),  
( 'stala@gmail.com', '777 357 007'),  
( 'vodova@gmail.com', '667 345 334'),  
( 'kolarova@gmail.com', '773 327 337'),  
( 'dobesova@gmail.com', '763 445 119'),  
( 'hotova@gmail.com', '765 333 056');
```

`insert into kontakt_otec values`

```
('novak@seznam.cz', '756 235 007'),  
( 'novotny@gmail.com', '753 229 876'),  
( 'staly@gmail.com', '680 150 007'),  
( 'voda@gmail.com', '667 345 334'),  
( 'kolar@gmail.com', '710 155 337'),  
( 'dobes@gmail.com', '686 331 119'),  
( 'hot@gmail.com', '601 156 056');
```

`insert into otec_ditete values`

```
('Jan', 'Novák', 1),  
( 'Tomáš', 'Novotný', 2),  
( 'Jidáš', 'Stálý', 3),  
( 'Kryštof', 'Voda', 4),  
( 'Matěj', 'Kolář', 5),  
( 'Jan', 'Dobeš', 6),  
( 'Tonda', 'Hot', 7);
```

`insert into matka_ditete values`

```
('Hana', 'Novákáková', 1),  
( 'Jana', 'Novotná', 2),  
( 'Tereza', 'Stálá', 3),  
( 'Anna', 'Vodová', 4),  
( 'Michaela', 'Kolářová', 5),  
( 'Daniela', 'Dobešová', 6),  
( 'Pavla', 'Hotová', 7);
```

insert into psc values

('600 02', 'Brno', 'Kozí', '12'),
('634 02', 'Brno', 'Úzká', '8'),
('635 00', 'Brno', 'Větrná', '32'),
('635 00', 'Brno', 'Nad Kašnou', '14'),
('635 00', 'Brno', 'Rerychova', '8'),
('633 00', 'Brno', 'Dlouhá', '14'),
('628 00', 'Hodonín', 'Kaštany', '30'),
('611 00', 'Pasohlávky', 'Vlhká', '20'),
('635 00', 'Brno', 'Větrná', '15'),
('634 02', 'Brno', 'Jírova', '12');

insert into platba values

('1', 'Hotovost'),
('1', 'Účet'),
('1', 'Šek'),
('1', 'Kartou'),
('1', 'Šek'),
('0', 'Hotovost'),
('0', 'Šek'),
('0', 'Kartou'),
('0', 'Účet');

insert into kontakt_zam values

('kniha@seznam.cz', '763 345 007'),
('draha@gmail.com', '753 229 876'),
('volna@gmail.com', '780 150 007'),
('kostar@gmail.com', '767 345 334'),
('chudovec@gmail.com', '623 155 337');

insert into opraveni values

('admin', 'veškerá práva'),
('bez prav', 'žádné práva'),
('uprava', 'prava na mazani a upravu'),
('cteni', 'prava na cteni'),
('vkladani', 'prava na vkladani');

insert into pozice values

('uklizečka', 'uklízí prostory dětské skupiny'),
('recepční', 'pracuje na recepci');

```
('vychovatelka' , 'pracuje s dětmi'),  
( 'správce' , 'provádí údržbu prostor a majetku'),  
( 'IT pracovník' , 'Spravuje informační systém a web'),  
( 'ředitelka' , 'řídí dětskou skupinu a zaměstnance');
```

insert into vyzvedavani values

```
('Hana' , 'Novakova'),  
( 'Zuzana' , 'Ticha'),  
( 'Ivana' , 'Drobáčová'),  
( 'Jana' , 'Němečková'),  
( 'Pavla' , 'Kropáčková'),  
( 'Jirka' , 'Zirek');
```

insert into alergie values

```
('ořechy'),  
( 'laktoza'),  
( 'bodnutí'),  
( 'pyl'),  
( 'citrusy');
```

insert into rodny_list values

```
('1'),  
( '0');
```

insert into ockovaci_prukaz values

```
('1'),  
( '0');
```

insert into obedy values

```
('1'),  
( '0');
```

insert into tydenni_dochazka values

```
('Pondělí - Pátek'),  
( 'Středa - Pátek'),  
( 'Pondělí - Úterý'),  
( 'Úterý - Pátek'),  
( 'Pondělí - Středa');
```


insert into pobočka values

('Zvoneček',3),
('Hříbeček',4);

insert into zamestnanec values

('Jana', 'Pitrochová', "", '963007/4231', '1996-07-30',1,2,1,1,1),
('Honza', 'Troška', "", '903007/4231', '1990-07-30',2,4,2,2,1),
('Zuzana', 'Volná', 'Ing', 'Dis', '765623/9876', '1976-06-23',3,6,1,3,1);

insert into delka_dochazky values

('2019-01-01', '08:00:00'),
('2019-01-01', '06:30:00'),
('2019-01-01', '05:00:00'),
('2019-01-02', '05:00:00'),
('2019-01-02', '08:00:00'),
('2019-01-02', '06:30:00'),
('2019-01-03', '05:00:00'),
('2019-01-03', '08:00:00'),
('2019-01-03', '06:30:00'),
('2019-01-04', '05:00:00'),
('2019-01-04', '08:00:00'),
('2019-01-04', '06:30:00'),
('2019-01-05', '08:00:00'),
('2019-01-05', '08:00:00'),
('2019-01-05', '08:00:00'),
('2019-01-05', '06:30:00'),
('2019-01-05', '06:30:00');

insert into klient values

('Vojta', 'Novák', '153007/4231', '2015-07-30',1,1,1,1,1),
('Honza', 'Stálý', '142007/4111', '2014-07-20',2,2,1,2,2),
('Dana', 'Stálá', '142008/4001', '2014-08-20',3,3,1,3,2),
('Honza', 'Voda', '151309/4861', '2015-09-13',4,4,1,4,3),
('Jana', 'Kolářová', '142003/4311', '2014-03-20',5,5,1,5,3),
('Zuzana', 'Dobešová', '161211/4941', '2016-11-12',6,6,1,6,1),
('VAŠEK', 'Hot', '141112/4111', '2014-12-11',7,7,1,7,2);

`insert into pocet_dochazky values`

(1,1),
(1,4),
(1,10),
(3,1),
(3,4),
(3,8),
(4,10),
(4,1),
(5,2),
(5,10),
(6,2),
(6,8),
(7,3),
(7,6),
(7,11);

`insert into prihlaska values`

('Vojta','Novák','153007/4231','2015-07-30','Hana',
'Novákáková','756 230 333','Jan', 'Novák','756 235 007',1,1,1,1,1,1,1,3,1),
('Honza','Stálý','142007/4111','2014-07-20','Tereza', 'Stálá','777 357 007',
'Jidáš', 'Stálý','680 150 007',2,2,2,3,1,1,1,2,1);